

User Manual

for S32K3XX ETH Driver

Document Number: UM34ETHASRR21-11 Rev0000R3.0.0 Rev. 1.0

1 Revision History	2
2 Introduction	3
2.1 Supported Derivatives	3
2.2 Overview	4
2.3 About This Manual	5
2.4 Acronyms and Definitions	6
2.5 Reference List	6
3 Driver	8
3.1 Requirements	8
3.2 Driver Design Summary	8
3.3 Hardware Resources	9
3.4 Deviations from Requirements	9
3.5 Driver Limitations	24
3.6 Driver usage and configuration tips	24
3.6.1 External data buffer usage for transmission and reception	24
3.6.2 Cache management feature prerequisites:	27
3.7 Runtime errors	27
3.7.1 Standardized Extended Production Errors	27
3.7.2 Vendor-Specific Extended Production Errors	28
3.8 Symbolic Names Disclaimer	28
4 Tresos Configuration Plug-in	29
4.1 Module Eth	33
4.2 Container EthGeneral	33
4.3 Parameter EthDevErrorDetect	34
4.4 Parameter EthGetDropCountApi	34
4.5 Parameter EthGetEtherStatsApi	34
4.6 Parameter EthGetCounterValuesApi	35
4.7 Parameter EthGetRxStatsApi	35
4.8 Parameter EthGetTxStatsApi	36
4.9 Parameter EthGetTxErrorCounterValuesApi	36
4.10 Parameter EthSendMultiBufferFrameApi	37
4.11 Parameter EthGlobalTimeSupport	37
4.12 Parameter EthVersionInfoApi	38
4.13 Parameter EthIndex	38
4.14 Parameter EthMainFunctionPeriod	39
4.15 Parameter EthMaxCtrlsSupported	39
4.16 Parameter EthMaxBuffersInMultiBufferFrame	40
4.17 Reference EthEcucPartitionRef	40

4.18 Container EthCtrlOffloading	41
4.19 Parameter EthCtrlEnableOffloadChecksumICMP	41
4.20 Parameter EthCtrlEnableOffloadChecksumIPv4	42
4.21 Parameter EthCtrlEnableOffloadChecksumTCP	42
4.22 Parameter EthCtrlEnableOffloadChecksumUDP	43
4.23 Container EthGeneralVendorSpecific	43
4.24 Parameter EthDisableDemEventDetect	43
4.25 Parameter EthEnableUserModeSupport	44
4.26 Parameter EthMulticoreSupport	44
4.27 Parameter EthUpdatePhysAddrFilterApi	45
4.28 Parameter EthSwtManagementSupportApi	45
4.29 Parameter EthCoalescingInterrupt	46
4.30 Parameter EthTimeoutMethod	46
4.31 Parameter EthTimeoutDuration	47
4.32 Parameter EthEnableCacheManagement	47
4.33 Container EthConfigSet	48
4.34 Container EthCtrlConfig	48
4.35 Parameter EthCtrlEnableMii	49
4.36 Parameter EthCtrlEnableMmd	49
4.37 Parameter EthCtrlEnableRxInterrupt	50
4.38 Parameter EthCtrlEnableTxInterrupt	50
4.39 Parameter EthCtrlIdx	51
4.40 Parameter EthCtrlMacLayerType	51
4.41 Parameter EthCtrlMacLayerSubType	52
4.42 Parameter EthCtrlMacLayerSpeed	52
4.43 Parameter EthCtrlPhyAddress	53
4.44 Parameter EthCtrlEnableSpiInterface	53
4.45 Parameter EthCtrlConfigSwBufferHandling	54
4.46 Reference EthCtrlEcucPartitionRef	54
4.47 Container EthCtrlVendorSpecific	55
4.48 Container EthCtrlConfigGeneral	55
4.49 Parameter EthCtrlEnableAtInit	56
4.50 Parameter EthCtrlAllocateTxDataBuffers	56
4.51 Parameter EthCtrlAllocateRxDataBuffers	57
4.52 Parameter EthCtrlEthIfIdx	57
4.53 Parameter EthEthTrcvDriverVendorIdAndApiInfix	58
4.54 Parameter EthEthSwtDriverVendorIdAndApiInfix	58
4.55 Parameter EthDuplexMode	59
4.56 Parameter EthTimeStampRequiredAccuracy	59
4.57 Reference EthModuleReferenceClock	60

4.58 Reference EthTimeStampReferenceClock	60
4.59 Container EthCtrlConfigSafety	61
4.60 Parameter EthCtrlSafetyCallback	61
4.61 Parameter EthCtrlEnableEcc	62
4.62 Parameter EthCtrlEnableDpp	62
4.63 Parameter EthCtrlEnableFsmSpp	63
4.64 Parameter EthCtrlEnableFsmTimeout	63
4.65 Reference ETH_E_SAFETY	64
4.66 Container EthCtrlConfigMac	64
4.67 Parameter MAC_CONFIG_CRC_STRIPPING	65
4.68 Parameter MAC_CONFIG_AUTO_PAD	65
4.69 Parameter MAC_CONFIG_JUMBO_PKT_EN	65
4.70 Parameter MAC_CONFIG_LOOPBACK	66
4.71 Parameter MAC_CONFIG_ENABLE_CRD_FD	66
4.72 Parameter MAC_CONFIG_DISABLE_RECEIVE_OWN_HD	67
4.73 Parameter MAC_CONFIG_DISABLE_CRD_HD	67
4.74 Parameter MAC_CONFIG_DISABLE_RETRY	68
4.75 Parameter MAC_CONFIG_DEFERRAL_CHECK_HD	68
4.76 Container EthCtrlConfigPacketFilter	69
4.77 Parameter PKT_FILTER_RECV_ALL	69
4.78 Parameter PKT_FILTER_HASH_OR_PERFECT_FILTER	70
4.79 Parameter PKT_FILTER_BLOCK_CONTROL_PKTS	70
4.80 Parameter PKT_FILTER_BLOCK_PAUSE_PKTS	71
4.81 Parameter PKT_FILTER_PASS_ALL_CONTROL_PKTS	71
4.82 Parameter PKT_FILTER_PASS_CONTROL_PKTS_ADDR_MATCH	72
4.83 Parameter PKT_FILTER_DISABLE_BROADCAST	72
4.84 Parameter PKT_FILTER_PASS_ALL_MULTICAST	72
4.85 Parameter PKT_FILTER_DST_ADDR_INV_FILTER_EN	73
4.86 Parameter PKT_FILTER_HASH_MULTICAST	73
4.87 Parameter PKT_FILTER_HASH_UNICAST	74
4.88 Parameter PKT_FILTER_PROMISCUOUS_MODE	74
4.89 Container EthCtrlConfigEgress	75
4.90 Reference EthCtrlConfigEgressLastSchedulerRef	75
4.91 Container EthCtrlConfigEgressFifo	76
4.92 Parameter EthCtrlConfigEgressFifoBufLenByte	76
4.93 Parameter EthCtrlConfigEgressFifoBufTotal	77
4.94 Parameter EthCtrlConfigEgressFifoIdx	77
4.95 Parameter EthCtrlConfigEgressFifoCallback	78
4.96 Parameter EthCtrlConfigEgressFifoPriorityAssignment	78
4.97 Container EthCtrlConfigScheduler	79

4.98 Container EthCtrlConfigSchedulerPredecessor	79
4.99 Parameter EthCtrlConfigSchedulerPredecessorOrder	80
4.100 Reference EthCtrlConfigSchedulerPredecessorRef	80
4.101 Container EthCtrlConfigSchedulerPredecessorVendorSpecific	81
4.102 Parameter EthQueueBandwidth	81
4.103 Container EthCtrlConfigSchedulerVendorSpecific	82
4.104 Parameter EthTxSchedulerAlgorithm	82
4.105 Parameter EthTotalPacketsPerCycle	83
4.106 Parameter EthTotalQuantumPerCycle	83
4.107 Container EthCtrlConfigShaper	84
4.108 Parameter EthCtrlConfigShaperIdleSlope	84
4.109 Parameter EthCtrlConfigShaperMaxCredit	85
4.110 Parameter EthCtrlConfigShaperMinCredit	85
4.111 Reference EthCtrlConfigShaperPredecessorFifoRef	86
4.112 Container EthCtrlConfigShaperVendorSpecific	86
4.113 Parameter EthCtrlConfigHiCredit	87
4.114 Parameter EthCtrlConfigLoCredit	87
4.115 Container EthCtrlConfigTimeAwareShaper	88
4.116 Container EthCtrlConfigTimeAwareShaper	88
4.117 Parameter EthGateConfigBaseTimeInSecond	88
4.118 Parameter EthGateConfigBaseTimeInNanoSecond	89
4.119 Parameter EthGateConfigCycleTime	90
4.120 Parameter EthGateConfigExtendTime	90
4.121 Parameter EthCtrlEnablePreemption	91
4.122 Parameter EthReleaseAdvanceTime	91
4.123 Parameter EthHoldAdvanceTime	92
4.124 Container EthGateControlList	92
4.125 Parameter EthCtrlConfigInterval	92
4.126 Container EthGateControlFifo	93
4.127 Parameter EthCtrlConfigGateStatus	93
4.128 Reference EthCtrlConfigFifoRef	94
4.129 Container EthPreemptionClassification	94
4.130 Parameter EthCtrlConfigPreemptionClassification	95
4.131 Reference EthPreemptionFifoRef	95
4.132 Container EthCtrlConfigIngress	96
4.133 Container EthCtrlConfigIngressFifo	96
4.134 Parameter EthCtrlConfigIngressFifoBufLenByte	97
4.135 Parameter EthCtrlConfigIngressFifoBufTotal	97
4.136 Parameter EthCtrlConfigIngressFifoIdx	98
4.137 Parameter EthCtrlConfigIngressFifoCallback	98

4.138 Parameter EthCtrlConfigIngressFifoPriorityAssignment	98
4.139 Container EthDemEventParameterRefs	99
4.140 Reference ETH_E_ACCESS	100
4.141 Reference ETH_E_RX_FRAMES_LOST	100
4.142 Reference ETH_E_CRC	101
4.143 Reference ETH_E_UNDERSIZEFRAME	101
4.144 Reference ETH_E_OVERSIZEFRAME	102
4.145 Reference ETH_E_ALIGNMENT	102
4.146 Reference ETH_E_SINGLECOLLISION	103
4.147 Reference ETH_E_MULTIPLECOLLISION	103
4.148 Reference ETH_E_LATECOLLISION	104
4.149 Container EthCtrlConfigSpiConfiguration	104
4.150 Parameter EthCtrlConfigSpiCommTimeout	104
4.151 Parameter EthCtrlConfigSpiChunkPayloadSize	105
4.152 Parameter EthCtrlConfigSpiCommRetries	106
4.153 Parameter EthCtrlConfigSpiEnableControlDataProtection	106
4.154 Parameter EthCtrlConfigSpiEnableRxCSAlign	107
4.155 Parameter EthCtrlConfigSpiEnableRxCutThrough	107
4.156 Parameter EthCtrlConfigSpiEnableRxZeroAlign	109
4.157 Parameter EthCtrlConfigSpiEnableTransmitDataHdrSequence	110
4.158 Parameter EthCtrlConfigSpiEnableTxChecksum	110
4.159 Parameter EthCtrlConfigSpiEnableTxCutThrough	111
4.160 Parameter EthCtrlConfigSpiSelectTimeStamp	111
4.161 Parameter EthCtrlConfigSpiTransmitCreditThreshold	112
4.162 Container EthCtrlConfigSpiSequence	112
4.163 Parameter EthCtrlConfigSpiAccessSynchronous	113
4.164 Reference EthCtrlConfigSpiSequenceName	114
4.165 Container CommonPublishedInformation	114
4.166 Parameter ModuleId	115
4.167 Parameter VendorId	115
4.168 Parameter VendorApiInfix	116
4.169 Parameter ArReleaseMajorVersion	116
4.170 Parameter ArReleaseMinorVersion	117
4.171 Parameter ArReleaseRevisionVersion	117
4.172 Parameter SwMajorVersion	118
4.173 Parameter SwMinorVersion	118
4.174 Parameter SwPatchVersion	119

5 Module Index 120

5.1 Software Specification	120
--------------------------------------	-----

6 Data Structure Index	121
6.1 Data Structures	121
7 Module Documentation	122
7.1 ETH Driver	122
7.2 GMAC Driver	123
7.2.1 Detailed Description	123
7.2.2 Data Structure Documentation	135
7.2.3 Macro Definition Documentation	145
7.2.4 Types Reference	148
7.2.5 Enum Reference	149
7.2.6 Function Reference	154
7.3 Ethernet Driver	177
7.3.1 Detailed Description	177
7.3.2 Macro Definition Documentation	177
7.3.3 Function Reference	178
8 Data Structure Documentation	184
8.1 Gmac_Ip_ManagementInfo Struct Reference	184
8.1.1 Detailed Description	184



Chapter 1

Revision History

Revision	Date	Author	Description
1.0	31.03.2023	NXP RTD Team	S32K3 Real-Time Drivers AUTOSAR 4.4 & R21-11 Version 3.0.0

Chapter 2

Introduction

- [Supported Derivatives](#)
- [Overview](#)
- [About This Manual](#)
- [Acronyms and Definitions](#)
- [Reference List](#)

This User Manual describes NXP Semiconductors' AUTOSAR Ethernet Driver for S32K3XX.

AUTOSAR Ethernet Driver configuration parameters description can be found in the Tressos Configuration Plugin section. Deviations from the specification are described in the [Deviations from Requirements](#) section.

AUTOSAR Ethernet driver requirements and APIs are described in the Ethernet Driver Software Specification Document.

2.1 Supported Derivatives

The software described in this document is intended to be used with the following microcontroller devices of NXP Semiconductors:

- s32k310_mqfp100
- s32k310_lqfp48
- s32k311_mqfp100 / MWCT2015S_mqfp100
- s32k311_lqfp48
- s32k312_mqfp100 / MWCT2016S_mqfp100
- s32k312_mqfp172 / MWCT2016S_mqfp172
- s32k314_mqfp172
- s32k314_mapbga257

- s32k322_mqfp100 / MWCT2D16S_mqfp100
- s32k322_mqfp172 / MWCT2D16S_mqfp172
- s32k324_mqfp172 / MWCT2D17S_mqfp172
- s32k324_mapbga257
- s32k341_mqfp100
- s32k341_mqfp172
- s32k342_mqfp100
- s32k342_mqfp172
- s32k344_mqfp172
- s32k344_mapbga257
- s32k394_mapbga289
- s32k396_mapbga289
- s32k358_mqfp172
- s32k358_mapbga289
- s32k328_mqfp172
- s32k328_mapbga289
- s32k338_mqfp172
- s32k338_mapbga289
- s32k348_mqfp172
- s32k348_mapbga289
- s32m274_lqfp64
- s32m276_lqfp64

All of the above microcontroller devices are collectively named as S32K3.

Note: MWCT part numbers contain NXP confidential IP for Qi Wireless Power.

2.2 Overview

AUTOSAR (AUTomotive Open System ARchitecture) is an industry partnership working to establish standards for software interfaces and software modules for automobile electronic control systems.

AUTOSAR:

- paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.
- is a strong global partnership that creates one common standard: "Cooperate on standards, compete on implementation".
- is a key enabling technology to manage the growing electrics/electronics complexity. It aims to be prepared for the upcoming technologies and to improve cost-efficiency without making any compromise with respect to quality.
- facilitates the exchange and update of software and hardware over the service life of the vehicle.

2.3 About This Manual

This Technical Reference employs the following typographical conventions:

- **Boldface** style: Used for important terms, notes and warnings.
- *Italic* style: Used for code snippets in the text. Note that C language modifiers such "const" or "volatile" are sometimes omitted to improve readability of the presented code.

Notes and warnings are shown as below:

Note

This is a note.

Warning

This is a warning

2.4 Acronyms and Definitions

Term	Definition
API	Application Programming Interface
AUTOSAR	Automotive Open System Architecture
DEM	Diagnostic Event Manager
DET	Default Error Tracer
ETH	Ethernet
ETHIF	Ethernet Interface
ETHTRCV	Ethernet Transceiver
ETHSWT	Ethernet Switch
MCU	Micro controller Unit
MII	Media Independent Interface
N/A	Not Available
RMII	Reduced Media Independent Interface
RAM	Random Access Memory

- The term "Ethernet Controller" is related to the hardware module providing the Ethernet functionality.
- The term "Ethernet Driver" is related to the software handling the Ethernet Controller.
- The term "Application" is used for the software utilizing the Ethernet Driver.

Term	Definition
API	Application Programming Interface
AUTOSAR	Automotive Open System Architecture
DEM	Diagnostic Event Manager
DET	Default Error Tracer
ETH	Ethernet
ETHIF	Ethernet Interface
ETHTRCV	Ethernet Transceiver
ETHSWT	Ethernet Switch
MCU	Microcontroller Unit
MII	Media Independent Interface
N/A	Not Available
RMII	Reduced Media Independent Interface
RGMII	Reduced Gigabit Media Independent Interface
RAM	Random Access Memory

- The term "Ethernet Controller" is related to the hardware module providing the Ethernet functionality.
- The term "Ethernet Driver" is related to the software handling the Ethernet Controller.
- The term "Application" is used for the software utilizing the Ethernet Driver.

2.5 Reference List

#	Title	Version
1	Specification of ETH Driver	AUTOSAR R21-11
2	Reference Manual	S32K3xx Reference Manual, Rev.6, Draft B, 01/2023
		S32K39 and S32K37 Reference Manual, Rev. 2 Draft A, 11/2022
3	Datasheet	S32K3xx Data Sheet, Rev. 6, 11/2022
		S32K396 Data Sheet, Rev. 1.1, 08/2022
4	Errata	S32K358_0P14E Mask Set Errata – Rev. 28, 9/2022
		S32K396_0P40E Mask Set Errata, Rev. DEC2022, 12/2022
		S32K311_0P98C Mask Set Errata, Rev. 6/March/2023, 3/2023
		S32K312 Mask Set Errata for Mask 0P09C, Rev. 25/April/2022
		S32K342 Mask Set Errata for Mask 0P97C, Rev. 10, 11/2022
		S32K3x4 Mask Set Errata for Mask 0P55A/1P55A, Rev. 14/Oct/2022

Chapter 3

Driver

- [Requirements](#)
- [Driver Design Summary](#)
- [Hardware Resources](#)
- [Deviations from Requirements](#)
- [Driver Limitations](#)
- [Driver usage and configuration tips](#)
- [Runtime errors](#)
- [Symbolic Names Disclaimer](#)

3.1 Requirements

Requirements for this driver are detailed in the AUTOSAR 4.4 & R21-11 Ethernet Driver Software Specification document (See Table [Reference List](#))

3.2 Driver Design Summary

The Ethernet Driver controls the Ethernet Media Access Controller (EMAC) or Gigabit Ethernet Media Access Controller (GMAC). It provides the following features:

- Configuration and initialization of the Ethernet Controller
- Switching the Ethernet Controller on and off
- Reception and transmission of Ethernet frames
- Access to the Ethernet Controller's RMON counters (RxStats, TxStats and TxErrorCounterValues)
- Access to the Ethernet Transceiver device registers through MDIO
- Handling of the Ethernet Controller's interrupt requests

- Half and full duplex operation support
- Various PHY Interfaces (10/100 Mbps on MII/RMII AND 125Mbps on RGMII)
- Time Synchronization over Ethernet (The gPTP stack has to be provided by the upper layers)
- Time Aware shaper, frame preemption for time sensitive networking.
- Checksum hardware offloading for IPv4/IPv6 with TCP, UDP or ICMP

3.3 Hardware Resources

For S32K3x1, S32K3x2, S32K3x4, S32K396, S32K394 (except S32K311/S32K312), The hardware module configured by the Ethernet driver is EMAC (Ethernet Media Access Controller). It has a hardware instance (EMAC_0). For S32K358, The hardware module configured by the Ethernet driver is GMAC (Gigabit Ethernet Media Access Controller). It has a single hardware instance (GMAC_0), this instance is with 1Gbps (1000 Mbps) support.

The Ethernet controller to microcontroller pin mapping can be done using the file **S32K342_S32K341_IOMUX.xlsx** or **S32K344_S32K324_S32K314_IOMUX.xlsx** or **S32K358_S32K348_S32K338_S32K328_IOMUX.xlsx** or **S32K39_and_S32K37_IOMUX.xlsx** attached to the Reference Manual.

3.4 Deviations from Requirements

The driver deviates from the AUTOSAR ETH Driver software specification in some places. The table below identifies the AUTOSAR requirements that are not implemented or out of scope for the ETH Driver.

Term	Definition
N/S	Out of scope
N/I	Not implemented
N/F	Not fully implemented

Below table identifies the AUTOSAR requirements that are not fully implemented, implemented differently or out of scope for the ETH driver.

Requirement	Status	Description	Notes
SWS_Eth_00263	N/S	If the configuration parameter EthCtrlConfigSwBufferHandling is set to TRUE, then the optional SW buffer handling shall be enabled.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00264	N/S	If the config parameter EthCtrlConfigSwBufferHandling is set to TRUE, then each SW FIFO shall have the total amount of elements given by EthCtrlConfigEgressFifoBufTotal (). Each element shall be of type EthCtrlBufIdxType.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress

Requirement	Status	Description	Notes
SWS_Eth_00265	N/S	"All SW FIFOs shall follow the criteria listed here: Each SW FIFO shall be filled and read out according to FIFO principles. The SW FIFOs shall support independent configuration regardless of any settings on the rest of SW FIFOs."	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00266	N/S	SW FIFOs shall be iterated and their credits account be updated in the following way and order: - Credits are only accumulated for SW FIFOs which have at least one message queued inside them. Empty SW FIFOs do not accumulate credits and their credits counter shall be set to 0. - Iterate through all SW FIFOs, starting at the highest priority SW FIFO and descending, and add the amount of credits accumulated since the last Eth_MainFunction() call. The amount of credits accumulated is given by Eth_CtrlConfigShaperIdleSlope. - If a SW FIFO reaches EthCtrlConfigShaperMaxCredit then the credit accumulation shall stop at that point and the next SW FIFO in the row is handled.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00267	N/S	If Eth_ProvideTxBuffer() is called and EthCtrlConfigSwBufferHandling is set to TRUE, a tuple of BuffIdx pointer to the SW buffer pool (which is returned) and priority (provided by argument of the current function call) shall be stored.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00268	N/S	When Eth_Transmit() is called and EthCtrlConfigSwBufferHandling is set to TRUE, the given BuffIdx pointer shall be assigned to the SW FIFO with the EthCtrlConfigEgressFifoPriorityAssignment which matches the priority given previously by the previous Eth_ProvideTxBuffer() call (see SWS_Eth_00267).	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00269	N/S	Upon calling Eth_Transmit(), messages from the SW FIFOs shall be moved to the HW FIFO as described in SWS_Eth_00271.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress

Requirement	Status	Description	Notes
SWS_Eth_00270	N/S	In the context of Eth_MainFunction(), the following actions shall be executed in the given order: - All SW FIFOs shall be iterated and their credits account updated as specified in SWS_Eth_00266. - All SW FIFOs shall be iterated and checked for messages which are ready for transmission. - For each SW FIFO iterated, transmission shall be attempted as specified in SWS_Eth_00271.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00271	N/S	Messages queued inside SW FIFOs shall be moved to the HW FIFO in the following way and order: - Loop through each SW FIFO, starting at the highest priority in descending order. - Move the first message inside a SW FIFO whose credit account is at least EthCtrlConfigShaperMinCredit to the HW FIFO. - If EthTrcvPhysLayerPLCAMaxBurstCount is set to 0 then only one message is moved to the HW FIFO and the iteration to the next SW FIFOs is stopped. - Reduce the SW FIFOs credits based on its EthCtrlConfigShaperSendSlope configuration. - If EthTrcvPhysLayerPLCAMaxBurstCount is higher than 0 then proceed on top as specified in SWS_Eth_00272.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00272	N/S	If EthTrcvPhysLayerPLCAMaxBurstCount is higher than 0, as many messages as EthTrcvPhysLayerPLCAMaxBurstCount indicates shall be moved additionally to the HW FIFO. The selection of each message shall be based on the requirements in SWS_Eth_00271.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00273	N/S	If Clause 45 registers need to be written via this access mechanism, the API shall use the register 13 and 14 to access them as explicitly specified by the annex 22D [20].	Propose to reject: We provide extension function for Clause45 support. Clause45 support through Clause22 operations can be done by the user.
SWS_Eth_00274	N/S	If Clause 45 registers need to be read via this access mechanism, the API shall use the register 13 and 14 to access them as explicitly specified by the annex 22D [20].	Propose to reject: We provide extension function for Clause45 support. Clause45 support through Clause22 operations can be done by the user.

Requirement	Status	Description	Notes
ECUC_Eth_00071	N/S	Name: EthCtrlConfigSwBufferHandling Parent Container: EthCtrlConfig Description: Enables / Disables SW buffer management Multiplicity: 1 Type: EcucBooleanParamDef Default value: – Post-Build Variant Value: false Value Configuration Class: Pre-compile time: X: All Variants Link time: – Post-build time: – Scope / Dependency: scope: local	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00285	N/S	If an off-chip Ethernet controller is used (i.e. MACPHY), the Ethernet controller module shall use services of other MCAL drivers (e.g. SPI).	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_00299	N/S	If the configuration parameter EthCtrlConfigSwBufferHandling is set to TRUE, then one SW FIFO shall be available per configured EthCtrlConfigEgressFifo.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00298	N/S	If the configuration parameter EthCtrlConfigSwBufferHandling is set to TRUE, then each SW FIFO shall handle frames according to the configured priorities given by EthCtrlConfigEgressFifoPriorityAssignment aggregated by the according EthCtrlConfigEgressFifo. If no EthCtrlConfigEgressFifoPriorityAssignment is configured, then any priority shall be handled by this SW FIFO.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00297	N/S	If the config parameter EthCtrlConfigSwBufferHandling is set to TRUE, then a SW buffer shall be provided with a size according to all configured EthCtrlConfigEgressFifo's. The size of each EthCtrlConfigEgressFifo shall be calculated in bytes by considering the following formula: size of one EthCtrlConfigEgressFifo = EthCtrlConfigEgressFifoBufTotal * EthCtrlConfigEgressFifoBufLenByte.	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00300	N/S	If the config parameter EthCtrlConfigSwBufferHandling is set to TRUE, then all SW FIFOs and SW buffer pools shall be initialized with '0'	SW buffer management is not needed, as the hardware already has hardware buffers for both egress and ingress
SWS_Eth_00287	N/S	If EthCtrlEnableSpiInterface is TRUE, the function shall process the write request as described in the TC6 [25].	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
SWS_Eth_00290	N/S	If EthCtrlEnableSpiInterface is TRUE, the function shall process the read request as described in the TC6 [25].	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_00292	N/S	If development error detection is enabled: the function shall check that the service Eth_Init was previously called. If the check fails, the function shall raise the development error ETH_E_UNINIT.	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_00293	N/S	If development error detection is enabled: the function shall check the parameter CtrlIdx for being valid. If the check fails, the function shall raise the development error ETH_E_INVALID_CTRL_IDX.	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_00294	N/S	If development error detection is enabled: the function shall check the parameter TxStats for being valid. If the check fails, the function shall raise the development error ETH_E_INVALID_RAM_POINTER.	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_00295	N/S	The function Eth_GetSpiStatus shall be pre compile time configurable On/Off by the configuration parameter: EthCtrlEnableSpiInterface.	No off-chip Ethernet controllers are in scope of this driver
SWS_Eth_91012	N/S	"Service Name: Eth_GetSpiStatus (draft) Syntax: Std_ReturnType Eth_GetSpiStatus (uint8 CtrlIdx, Eth_SpiStatusType* SpiStatusType) Service ID [hex]: 0x1E Sync/Async: Synchronous Reentrancy: Non Reentrant Parameters (in): CtrlIdx: Index of the controller within the context of the Ethernet Driver Parameters (inout): None Parameters (out): SpiStatusType: MACPHY status Return value: Std_ReturnType: E_OK: success, E_NOT_OK: Status could not be obtained Description: Returns the status defined by OA TC6 [26] to identify if an error can occurred at the SPI interface.: Tags: atp.Status=draft Available via: Eth.h"	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
SWS_Eth_91013	N/S	<p>Name: Eth_SpiStatusType (draft) Kind: Structure Elements: Spi← StatusRegister Type: uint32 Comment: Bit mapped status defined by OA TC6 [26] to notify following information:: (Pos : description)← : 0x00: Transmit_Protocol_← Error,; 0x01: Transmit_Buffer_← Overflow_Error,; 0x02: Transmit← _Buffer_Underflow_Error,; 0x03: Receive_Buffer_Overflow_Error,; 0x04: Loss_Framing_error,; 0x05: Header_Error,; 0x06: Reset_← Complete,; 0x07: PHY_Interrupt,← : 0x08: Transmit_Timestamp Capture_Available_A,; 0x09← : Transmit_Timestamp_Capture← _Available_B,; 0x0A: Transmit_← Timestamp_Capture_Available_C,; 0x0B: Transmit_Frame_Check_← Sequence_Error,; 0x0C: Control← _Data_Protection_Error,; 0x0D - 0xFF: Reserved. Sync Type: boolean Comment: Synchronization configura- tion as defined in the OA TC6 [26]. TRUE: MACPHY has been reset and is not configured. FALSE: MACPHY is configured. BufferStatusTxCredit Type: uint8 Comment: Contains the number of consecutive transmitted data chunks of Ethernet frame the SPI host can write without overflow- ing the MAC. BufferStatusRxCredit Type: uint8 Comment: Contains the number of additional received data chunks of Ethernet frame currently available for the SPI host to read. Description: Returns the Spi status, errors and configuration state.: Tags: atp.Status=draft Available via: Eth.h</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00073	N/S	"Name: EthCtrlEnableSpiInterface Parent Container: EthCtrlConfig Description: This optional parameter enables the processing of control data and Ethernet frames over the SPI interface specific for MACPHY device. The use of this parameter implies the respect of the SPI protocol described in TC6 [26]. Multiplicity: 0..1 Type: EcucBooleanParamDef Default value: false Post-Build Variant Multiplicity: false Post-Build Variant Value: true Value Configuration Class: Pre-compile time: X: All Variants Link time: -- Post-build time: -- Scope / Dependency: scope: local"	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00074	N/S	"Container Name: EthCtrlConfig SpiConfiguration Parent Container: EthCtrlConfig Description: SPI Interface configuration of one Ethernet controller (MACPHY use). Configured only if EthCtrlEnableSpiInterface is set to TRUE.: Tags: atp.Status=draft Post-Build Variant Multiplicity: false Configuration Parameters"	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00079	N/S	"Name: EthCtrlConfigSpiChunkPayloadSize Parent Container: EthCtrlConfigSpiConfiguration Description: Configures the size of the payload chunks which will be transferred over the SPI interface. Note: The chunk is the basic element for data transaction over the SPI which can be a section of an Ethernet frame or management command. The configured value has to be a multiple of 8.: Tags: atp.Status=draft Multiplicity: 1 Type: EcucIntegerParamDef Range: 8 .. 64 Default value: 64 Post-Build Variant Value: true Value Configuration Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE."	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00075	N/S	<p>"Name: EthCtrlConfigSpiComm↵ Retries Parent Container: EthCtrl↵ ConfigSpiConfiguration Description: Indicates the maximum number of communication retries in case of a failed SPI communication (applies both to timed out communication and to errors/NACK in the response data). If configured value is '0', no retry is allowed (communication is expected to succeed at first try).: Tags: atp.↵ Status=draft Multiplicity: 1 Type: EcucIntegerParamDef Range: 0 .. 255 Default value: -- Post-Build Variant Value: true Value Configuration Class: Pre-compile time: X: VARI↵ ANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE. This parameter exists only if at least one SPI Sequence is referenced."</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00076	N/S	<p>"Name: EthCtrlConfigSpiComm↵ Timeout Parent Container: EthCtrl↵ ConfigSpiConfiguration Description: Indicates the maximum time allowed to the Ethernet controller for replying (either positively or negatively) to a SPI command. Timeout is configured in seconds. Timeout value of '0' means that no specific timeout is to be used by Ethernet controller and the communication is executed at the best of the SPI HW capacity.: Tags: atp.↵ Status=draft Multiplicity: 1 Type: EcucFloatParamDef Range: [0 .. 0.1] Default value: -- Post-Build Variant Value: true Multiplicity Configuration Class: Pre-compile time: -- Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-↵ POST-BUILD Value Configuration Class: Pre-compile time: X: VARI↵ ANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE. This parameter exists only if at least one SPI Sequence is referenced."</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00081	N/S	"Name: EthCtrlConfigSpiEnable↵ ControlDataProtection Parent Container: EthCtrlConfigSpi↵ Configuration Description: Enables the control data protection. When set, all control data written to and read from the MACPHY will be transferred with its complement for detection of bit errors as defined in OA TC6 [26]. FALSE: Control data read/write protection is disabled (unprotected). TRUE: Control data read/write protection is enabled (protected).: Tags: atp.Status=draft Multiplicity: 1 Type: EcucBooleanParamDef Default value: false Post-Build Variant Value: false Value Configuration Class: Pre-compile time: X: All Variants Link time: -- Post-build time: -- Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00085	N/S	"Name: EthCtrlConfigSpiEnableRx↵ CSAlign Parent Container: EthCtrl↵ ConfigSpiConfiguration Description↵ : Configures the CSn Align Receive frame. TRUE: all received Ethernet frames data shall start at the beginning of the first receive data chunk payload following CSn assertion FALSE↵ E: received frames may begin within any receive data chunk of the transaction when this bit is clear.: Tags↵ : atp.Status=draft Multiplicity: 1 Type: EcucBooleanParamDef Default value: false Post-Build Variant Value↵ : true Value Configuration Class↵ : Pre-compile time: X: VARIAN↵ NT-PRE-COMPILE Link time: X↵ : VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUIL↵ D Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00082	N/S	<p>"Name: EthCtrlConfigSpiEnable↵ RxCutThrough Parent Container↵ : EthCtrlConfigSpiConfiguration Description: When supported by the HW, enables the cut through mode of frame from the network to the SPI host.: Tags: atp.Status=draft Multiplicity: 0..1 Type: Ecuc↵ BooleanParamDef Default value: false Post-Build Variant Multiplicity: true Post-Build Variant Value: true Multiplicity Configuration Class: Pre-compile time: X: VARIAN↵ T-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-B↵ UILD Value Configuration Class: Pre-compile time: X: VARIAN↵ T-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00084	N/S	<p>"Name: EthCtrlConfigSpiEnable↵ RxZeroAlign Parent Container↵ : EthCtrlConfigSpiConfiguration Description: Configures the zero-align receive frame. TRUE: all received Ethernet frames data shall be aligned to start at the beginning of any receive data chunk payload. FALSE: Received frames may begin anywhere within the receive data chunk payload.: Tags: atp.Status=draft Multiplicity: 1 Type: EcucBooleanParamDef De- fault value: false Post-Build Variant Value: true Value Configuration Class: Pre-compile time: X: VARI↵ ANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00080	N/S	<p>"Name: EthCtrlConfigSpiEnable↵ TransmitDataHdrSequence Parent Container: EthCtrlConfigSpi↵ Configuration Description: When supported by the HW, enables the transmit data sequence monitor- ing. FALSE: transmit data header sequence bit monitoring disabled. TRUE: transmit data header sequence bit monitoring enabled.: Tags↵ : atp.Status=draft Multiplicity: 0..1 Type: EcucBooleanParamDef De- fault value: false Post-Build Variant Multiplicity: true Post-Build Variant Value: true Multiplicity Configura- tion Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-↵ POST-BUILD Value Configuration Class: Pre-compile time: X: VARI↵ ANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00086	N/S	<p>"Name: EthCtrlConfigSpiEnable↵ TxChecksum Parent Container↵ : EthCtrlConfigSpiConfiguration Description: Configures the CSn Align Receive frame. TRUE: all received Ethernet frames data shall start at the beginning of the first receive data chunk payload following CSn assertion FALSE: received frames may begin within any receive data chunk of the transaction when this bit is clear.: Tags: atp.Status=draft Multiplicity: 0..1 Type: EcucBooleanParamDef Default value: false Post-Build Vari- ant Multiplicity: true Post-Build Variant Value: true Multiplicity Con- figuration Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-↵ POST-BUILD Value Configuration Class: Pre-compile time: X: VARI↵ ANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local↵ : dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00089	N/S	<p>"Name: EthCtrlConfigSpiEnable↵ TxCutThrough Parent Container↵ : EthCtrlConfigSpiConfiguration Description: When supported by the HW, enables the cut through mode of frame from SPI host to the network.: Tags: atp.Status=draft Multiplicity: 0..1 Type: EcucBooleanParamDef Default value: false Post-Build Vari- ant Multiplicity: false Post-Build Variant Value: false Multiplicity Configuration Class: Pre-compile time: X: All Variants Link time: -- Post-build time: -- Value Configu- ration Class: Pre-compile time: X: All Variants Link time: -- Post-build time: -- Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00087	N/S	<p>"Name: EthCtrlConfigSpiSelectTimestamp Parent Container: EthCtrlConfigSpiConfiguration Description: When timestamp supported by the HW, selects size and format of the timestamps. FALSE: 32-bits timestamps TRUE: 64-bit timestamps Tags: atp.Status=draft Multiplicity: 0..1 Type: EcucBooleanParamDef Default value: false Post-Build Variant Multiplicity: true Post-Build Variant Value: true Multiplicity Configuration Class: Pre-compile time: X: VARIANT-T-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Value Configuration Class: Pre-compile time: X: VARIANT-T-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"</p>	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00083	N/S	<p>"Name: EthCtrlConfigSpiTransmitCreditThreshold Parent Container: EthCtrlConfigSpiConfiguration Description: Configures the minimum of available transmit credit before the writing IRQn is asserted. As per OA TC6, this information is notified by the TXC field. 0 = 1 credit 1 = 4 credits 2 = 8 credits 3 = 16 credits Tags: atp.Status=draft Multiplicity: 1 Type: EcucIntegerParamDef Range: 0 .. 3 Default value: 0 Post-Build Variant Value: true Value Configuration Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE."</p>	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00077	N/S	"Container Name: EthCtrlConfigSpiSequence Parent Container: EthCtrlConfigSpiConfiguration Description: Container gives Ethernet controller driver information about one SPI sequence. One SPI sequence used by Ethernet controller driver is in exclusive use for it. No other driver is allowed to access this sequence. Ethernet controller driver may use one sequence to access n Ethernet controller hardware chips of the same type or n sequences are used to access one single Ethernet controller hardware chip. If a Ethernet controller hardware has no SPI interface, there is no instance of this container.: Tags: atp.Status=draft Configuration Parameters"	No off-chip Ethernet controllers are in scope of this driver
ECUC_Eth_00078	N/S	"Name: EthCtrlConfigSpiAccess Synchronous Parent Container: EthCtrlConfigSpiSequence Description: This parameter is used to define whether the access to the Spi sequence is synchronous or asynchronous. true: SPI access is synchronous. false: SPI access is asynchronous.: Tags: atp.Status=draft Multiplicity: 0..1 Type: EcucBooleanParamDef Default value: false Post-Build Variant Multiplicity: true Post-Build Variant Value: true Multiplicity Configuration Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Value Configuration Class: Pre-compile time: X: VARIANT-PRE-COMPILE Link time: X: VARIANT-LINK-TIME Post-build time: X: VARIANT-POST-BUILD Scope / Dependency: scope: local: dependency: This parameter is valid, if EthCtrlEnableSpiInterface is configured and set to TRUE"	No off-chip Ethernet controllers are in scope of this driver

Requirement	Status	Description	Notes
ECUC_Eth_00088	N/S	"Name: EthCtrlConfigSpiSequence Name Parent Container: EthCtrlConfigSpiSequence Description: Reference to a Spi sequence configuration container.: Tags: atp.Status=draft Multiplicity: 0..* Type: Symbolic name reference to [SpiSequence] Post-Build Variant Multiplicity: false Post-Build Variant Value: false Multiplicity Configuration Class: Pre-compile time: X: All Variants Link time: - - Post-build time: -- Value Configuration Class: Pre-compile time: X: All Variants Link time: -- Post-build time: -- Scope / Dependency: scope: local: dependency: SpiSequence"	No off-chip Ethernet controllers are in scope of this driver

3.5 Driver Limitations

The Ethernet Driver has the following limitations:

- The API functions *Eth_43_GMAC_ReadMii* and *Eth_43_GMAC_WriteMii* (and their counterparts *Eth_43_GMAC_ReadMmd* and *Eth_43_GMAC_WriteMmd*) are synchronous instead of asynchronous due to a variety of reasons further detailed in [this](#) AUTOSAR ticket.
- The length of a single received frame (including the 14-bytes Ethernet frame header and the 4-bytes FCS) must be less than or equal to `EthCtrlConfigIngressFifoBufLenByte`.
- When preemption enabled, MMC can incorrectly operate for Tx when there are so many intermediate fragment frames that $SMD_Sx + MCRC_0 + SMD_Cx_1 + FragCount_1 + MCRC_1 + \dots + SMD_Cx_n + FragCount_n + FCS$ would make the frame transmission time to exceed the jabber limit (i.e. the time limit to transmit an uninterrupted/unfragmented frame with the maximum configured MTU). This is due to Errata ERR050597 can be find in Errata documentation in [<Reference List section>](#).

3.6 Driver usage and configuration tips

3.6.1 External data buffer usage for transmission and reception

Ethernet driver allows the usage of data buffers which are allocated externally in the application. This approach removes the data copy from the external buffers to the internal buffers which are by default used by the driver for transfers. In order to enable the External data buffers for Ethernet drive, the following configurations are necessary:

- Choose which data buffers are defined externally: data buffers for TX, RX or both and disable in configuration the following nodes:

1. `EthConfigSet/EthCtrlConfig/*/EthCtrlVendorSpecific/EthCtrlConfigGeneral/ EthCtrlAllocateTxDataBuffers` ,
 2. `EthConfigSet/EthCtrlConfig/*/EthCtrlVendorSpecific/EthCtrlConfigGeneral/ EthCtrlAllocateRxDataBuffers` ,
 3. both respectively.
- Allocate the data buffers at application level. The application must ensure that the address is aligned to 64 bytes. `VAR_ALIGN(address, size)` macro from RTD package can be used.
 - The data buffers must be placed in a memory section. There are two options: cacheable and no-cacheable.
 1. **cache management enabled** `EthGeneral/EthGeneralVendorSpecific/EthEnableCacheManagement` node is checked, meaning that the driver ensures cache management for the data buffers. In this case, data buffers must be placed in cacheable memory section delimited by `ETH_START_SEC_VAR_CLEARED_UNSPECIFIED` and `ETH_STOP_SEC_VAR_CLEARED_UNSPECIFIED` .
 2. **cache management disabled** `EthGeneral/EthGeneralVendorSpecific/EthEnableCacheManagement` node is not checked. In this case, data buffers must be placed in no cacheable memory section delimited by `ETH_START_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE` and `ETH_STOP_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE` .

3.6.1.1 Code snippets for External Buffers feature

The following code snippet illustrates how to allocate external data buffers in an application using both HLD Ethernet driver and IP Gmac driver. The data buffers are also placed in a no-cacheable or cacheable memory section depending on the macro `Gmac_HAS_CACHE_MANAGEMENT` which is generated based on the configuration mentioned above.

```
#if (GMAC_HAS_CACHE_MANAGEMENT == STD_ON)

#define ETH_START_SEC_VAR_CLEARED_UNSPECIFIED

#include "Eth_MemMap.h"

#else

#define ETH_START_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE

#include "Eth_MemMap.h"

#endif

VAR_ALIGN(uint8 RxFrameBuffer[GMAC_0_MAX_RXBUFF_SUPPORTED * GMAC_0_MAX_RXBUFFLEN_SUPPORTED *
FEATURE_GMAC_BUFFDESCR_ALIGNMENT_BYTES])

VAR_ALIGN(uint8 FrameBuffer[GMAC_0_MAX_TXBUFF_SUPPORTED * GMAC_0_MAX_TXBUFFLEN_SUPPORTED *
FEATURE_GMAC_BUFFDESCR_ALIGNMENT_BYTES])

#if (GMAC_HAS_CACHE_MANAGEMENT == STD_ON)

#define ETH_STOP_SEC_VAR_CLEARED_UNSPECIFIED
```

```
#include "Eth_MemMap.h"

#else

#define ETH_STOP_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE

#include "Eth_MemMap.h"

#endif
```

The next code snippet presents how to use the HLD driver APIs with External Buffer feature. The following usecase has the loopback mode enabled and one frame is sent using polling mode from EthController is received back on the same controller.

```
int main()
{
    int i;

    Std_ReturnType returnStatus;

    uint16 BufferLength;

    Eth_RxStatusType Status;

    /* 1. Call the init functions for the prerequisite modules and for eth driver*/

    /* 2. Set the external RX buffers addresses inside module descriptors for all the rings .

    This must be done always before enabling the controller, otherwise there could be frame loss and memory corruption
    due to missing data buffers on the reception.*/

    for (i=0; i < GMAC_0_MAX_RXBUFF_SUPPORTED; i++)
    {
        Eth_ProvideRxBuffer(EthController, i, RxFrameBuffer + i*GMAC_0_MAX_RXBUFFLEN_SUPPORTED);
    }

    /* 3. Enable the controller */

    /* 4. For TX data buffers the complete frame must contain: ETHERNET HEADER (DST MAC + SRC MAC +
    FRAME TYPE) + DATA PAYLOAD:

    The information must be copied in the buffer by the application before sending a frame. */

    /* 5. Call the Eth_SendFrame to send a buffer */

    returnStatus = Eth_SendFrame(EthController, Ring, TxFrameBuffer, &BufferLength, TRUE);

    /* 6. Wait frames to be transmitted. */

    Eth_TxConfirmation(EthController);
```



```

/* 7. Wait for frames to be received back. */
do
{
    Eth_Receive(EthController, 0U, &Status);
} while (Status == ETH_NOT_RECEIVED);
}

```

For the Gmac IP driver, the same code flow must be used, but the functions used are the followings (in order):

1. void Gmac_Ip_SetRxExternalBuffer(uint8 Instance, uint8 Ring, const Gmac_Ip_BufferType * Buff);
2. Gmac_Ip_StatusType Gmac_Ip_SendFrame(uint8 Instance, uint8 Ring, const Gmac_Ip_BufferType * Buff, const Gmac_Ip_RxInfo * RxInfo);
3. Gmac_Ip_StatusType Gmac_Ip_GetTransmitStatus(uint8 Instance, uint8 Ring, const Gmac_Ip_BufferType * Buff, Gmac_Ip_TransmitInfo * TxInfo);
4. Gmac_Ip_StatusType Gmac_Ip_ReadFrame(uint8 Instance, uint8 Ring, Gmac_Ip_BufferType * Buff, Gmac_Ip_RxInfo * RxInfo) and void Gmac_Ip_ProvideRxBuff(uint8 Instance, uint8 Ring, const Gmac_Ip_BufferType * Buff). In the last parameter of Gmac_Ip_ProvideRxBuff must be the Buff parameter returned from Gmac_Ip_ReadFrame.

Note

The detailed descriptions for the data types, driver functions can be found in the Module Documentation chapter of this manual.

The proper infix must be used for the HLD driver APIs, macros, data types and memory sections

3.6.2 Cache management feature prerequisites:

- Make sure the stack is put in non cacheable memory section. The used linker must be checked.

3.7 Runtime errors

The Ethernet driver generates the following DEM extended production errors at runtime:

3.7.1 Standardized Extended Production Errors

Function	Error code	Condition triggering the error
Eth_43_GMAC_MainFunction	ETH_E_ACCESS	Controller access failed
Eth_43_GMAC_MainFunction	ETH_E_RX_FRAMES_LOST	Rx frame lost detected
Eth_43_GMAC_MainFunction	ETH_E_CRC	CRC failure detected
Eth_43_GMAC_MainFunction	ETH_E_UNDERSIZEFRAME	Undersized frame detected
Eth_43_GMAC_MainFunction	ETH_E_OVERSIZEFRAME	Oversized frame detected
Eth_43_GMAC_MainFunction	ETH_E_ALIGNMENT	Alignment error detected
Eth_43_GMAC_MainFunction	ETH_E_SINGLECOLLISION	Single collision detected
Eth_43_GMAC_MainFunction	ETH_E_MULTIPLECOLLISION	Multiple collision detected
Eth_43_GMAC_MainFunction	ETH_E_LATECOLLISION	Late collision detected

Function	Error code	Condition triggering the error
Eth_43_GMAC_Ipw_SafetyIrqCallback	ETH_E_SAFETY	ECC error, datapath parity error, FSM state parity error, FSM timeout error

3.7.2 Vendor-Specific Extended Production Errors

3.8 Symbolic Names Disclaimer

All containers having symbolicNameValue set to TRUE in the AUTOSAR schema will generate defines like:

```
#define <Mip>Conf_<Container_ShortName>_<Container_ID>
```

For this reason it is forbidden to duplicate the names of such containers across the RTD configurations or to use names that may trigger other compile issues (e.g. match existing `#ifdefs` arguments).

Chapter 4

Tresos Configuration Plug-in

This chapter describes the Tresos configuration plug-in for the driver. All the parameters are described below.

- Module [Eth](#)
 - Container [EthGeneral](#)
 - * Parameter [EthDevErrorDetect](#)
 - * Parameter [EthGetDropCountApi](#)
 - * Parameter [EthGetEtherStatsApi](#)
 - * Parameter [EthGetCounterValuesApi](#)
 - * Parameter [EthGetRxStatsApi](#)
 - * Parameter [EthGetTxStatsApi](#)
 - * Parameter [EthGetTxErrorCounterValuesApi](#)
 - * Parameter [EthSendMultiBufferFrameApi](#)
 - * Parameter [EthGlobalTimeSupport](#)
 - * Parameter [EthVersionInfoApi](#)
 - * Parameter [EthIndex](#)
 - * Parameter [EthMainFunctionPeriod](#)
 - * Parameter [EthMaxCtrlsSupported](#)
 - * Parameter [EthMaxBuffersInMultiBufferFrame](#)
 - * Reference [EthEcucPartitionRef](#)
 - * Container [EthCtrlOffloading](#)
 - Parameter [EthCtrlEnableOffloadChecksumICMP](#)
 - Parameter [EthCtrlEnableOffloadChecksumIPv4](#)
 - Parameter [EthCtrlEnableOffloadChecksumTCP](#)
 - Parameter [EthCtrlEnableOffloadChecksumUDP](#)
 - * Container [EthGeneralVendorSpecific](#)
 - Parameter [EthDisableDemEventDetect](#)
 - Parameter [EthEnableUserModeSupport](#)
 - Parameter [EthMulticoreSupport](#)
 - Parameter [EthUpdatePhysAddrFilterApi](#)
 - Parameter [EthSwtManagementSupportApi](#)
 - Parameter [EthCoalescingInterrupt](#)
 - Parameter [EthTimeoutMethod](#)

- Parameter [EthTimeoutDuration](#)
- Parameter [EthEnableCacheManagement](#)
- Container [EthConfigSet](#)
 - * Container [EthCtrlConfig](#)
 - Parameter [EthCtrlEnableMii](#)
 - Parameter [EthCtrlEnableMmd](#)
 - Parameter [EthCtrlEnableRxInterrupt](#)
 - Parameter [EthCtrlEnableTxInterrupt](#)
 - Parameter [EthCtrlIdx](#)
 - Parameter [EthCtrlMacLayerType](#)
 - Parameter [EthCtrlMacLayerSubType](#)
 - Parameter [EthCtrlMacLayerSpeed](#)
 - Parameter [EthCtrlPhyAddress](#)
 - Parameter [EthCtrlEnableSpiInterface](#)
 - Parameter [EthCtrlConfigSwBufferHandling](#)
 - Reference [EthCtrlEcucPartitionRef](#)
 - Container [EthCtrlVendorSpecific](#)
 - Container [EthCtrlConfigGeneral](#)
 - Parameter [EthCtrlEnableAtInit](#)
 - Parameter [EthCtrlAllocateTxDataBuffers](#)
 - Parameter [EthCtrlAllocateRxDataBuffers](#)
 - Parameter [EthCtrlEthIfIdx](#)
 - Parameter [EthEthTrevDriverVendorIdAndApiInfix](#)
 - Parameter [EthEthSwtDriverVendorIdAndApiInfix](#)
 - Parameter [EthDuplexMode](#)
 - Parameter [EthTimeStampRequiredAccuracy](#)
 - Reference [EthModuleReferenceClock](#)
 - Reference [EthTimeStampReferenceClock](#)
 - Container [EthCtrlConfigSafety](#)
 - Parameter [EthCtrlSafetyCallback](#)
 - Parameter [EthCtrlEnableEcc](#)
 - Parameter [EthCtrlEnableDpp](#)
 - Parameter [EthCtrlEnableFsmSpp](#)
 - Parameter [EthCtrlEnableFsmTimeout](#)
 - Reference [ETH_E_SAFETY](#)
 - Container [EthCtrlConfigMac](#)
 - Parameter [MAC_CONFIG_CRC_STRIPPING](#)
 - Parameter [MAC_CONFIG_AUTO_PAD](#)
 - Parameter [MAC_CONFIG_JUMBO_PKT_EN](#)
 - Parameter [MAC_CONFIG_LOOPBACK](#)
 - Parameter [MAC_CONFIG_ENABLE_CRS_FD](#)
 - Parameter [MAC_CONFIG_DISABLE_RECEIVE_OWN_HD](#)
 - Parameter [MAC_CONFIG_DISABLE_CRS_HD](#)
 - Parameter [MAC_CONFIG_DISABLE_RETRY](#)
 - Parameter [MAC_CONFIG_DEFERRAL_CHECK_HD](#)
 - Container [EthCtrlConfigPacketFilter](#)
 - Parameter [PKT_FILTER_RECV_ALL](#)

- Parameter [PKT_FILTER_HASH_OR_PERFECT_FILTER](#)
- Parameter [PKT_FILTER_BLOCK_CONTROL_PKTS](#)
- Parameter [PKT_FILTER_BLOCK_PAUSE_PKTS](#)
- Parameter [PKT_FILTER_PASS_ALL_CONTROL_PKTS](#)
- Parameter [PKT_FILTER_PASS_CONTROL_PKTS_ADDR_MATCH](#)
- Parameter [PKT_FILTER_DISABLE_BROADCAST](#)
- Parameter [PKT_FILTER_PASS_ALL_MULTICAST](#)
- Parameter [PKT_FILTER_DST_ADDR_INV_FILTER_EN](#)
- Parameter [PKT_FILTER_HASH_MULTICAST](#)
- Parameter [PKT_FILTER_HASH_UNICAST](#)
- Parameter [PKT_FILTER_PROMISCUOUS_MODE](#)
- Container [EthCtrlConfigEgress](#)
- Reference [EthCtrlConfigEgressLastSchedulerRef](#)
- Container [EthCtrlConfigEgressFifo](#)
- Parameter [EthCtrlConfigEgressFifoBufLenByte](#)
- Parameter [EthCtrlConfigEgressFifoBufTotal](#)
- Parameter [EthCtrlConfigEgressFifoIdx](#)
- Parameter [EthCtrlConfigEgressFifoCallback](#)
- Parameter [EthCtrlConfigEgressFifoPriorityAssignment](#)
- Container [EthCtrlConfigScheduler](#)
- Container [EthCtrlConfigSchedulerPredecessor](#)
- Parameter [EthCtrlConfigSchedulerPredecessorOrder](#)
- Reference [EthCtrlConfigSchedulerPredecessorRef](#)
- Container [EthCtrlConfigSchedulerPredecessorVendorSpecific](#)
- Parameter [EthQueueBandwidth](#)
- Container [EthCtrlConfigSchedulerVendorSpecific](#)
- Parameter [EthTxSchedulerAlgorithm](#)
- Parameter [EthTotalPacketsPerCycle](#)
- Parameter [EthTotalQuantumPerCycle](#)
- Container [EthCtrlConfigShaper](#)
- Parameter [EthCtrlConfigShaperIdleSlope](#)
- Parameter [EthCtrlConfigShaperMaxCredit](#)
- Parameter [EthCtrlConfigShaperMinCredit](#)
- Reference [EthCtrlConfigShaperPredecessorFifoRef](#)
- Container [EthCtrlConfigShaperVendorSpecific](#)
- Parameter [EthCtrlConfigHiCredit](#)
- Parameter [EthCtrlConfigLoCredit](#)
- Container [EthCtrlConfigTimeAwareShaper](#)
- Container [EthCtrlConfigTimeAwareShaper](#)
- Parameter [EthGateConfigBaseTimeInSecond](#)
- Parameter [EthGateConfigBaseTimeInNanoSecond](#)
- Parameter [EthGateConfigCycleTime](#)
- Parameter [EthGateConfigExtendTime](#)
- Parameter [EthCtrlEnablePreemption](#)
- Parameter [EthReleaseAdvanceTime](#)
- Parameter [EthHoldAdvanceTime](#)
- Container [EthGateControlList](#)

- Parameter [EthCtrlConfigInterval](#)
- Container [EthGateControlFifo](#)
- Parameter [EthCtrlConfigGateStatus](#)
- Reference [EthCtrlConfigFifoRef](#)
- Container [EthPreemptionClassification](#)
- Parameter [EthCtrlConfigPreemptionClassification](#)
- Reference [EthPreemptionFifoRef](#)
- Container [EthCtrlConfigIngress](#)
- Container [EthCtrlConfigIngressFifo](#)
- Parameter [EthCtrlConfigIngressFifoBufLenByte](#)
- Parameter [EthCtrlConfigIngressFifoBufTotal](#)
- Parameter [EthCtrlConfigIngressFifoIdx](#)
- Parameter [EthCtrlConfigIngressFifoCallback](#)
- Parameter [EthCtrlConfigIngressFifoPriorityAssignment](#)
- Container [EthDemEventParameterRefs](#)
- Reference [ETH_E_ACCESS](#)
- Reference [ETH_E_RX_FRAMES_LOST](#)
- Reference [ETH_E_CRC](#)
- Reference [ETH_E_UNDERSIZEFRAME](#)
- Reference [ETH_E_OVERSIZEFRAME](#)
- Reference [ETH_E_ALIGNMENT](#)
- Reference [ETH_E_SINGLECOLLISION](#)
- Reference [ETH_E_MULTIPLECOLLISION](#)
- Reference [ETH_E_LATECOLLISION](#)
- Container [EthCtrlConfigSpiConfiguration](#)
- Parameter [EthCtrlConfigSpiCommTimeout](#)
- Parameter [EthCtrlConfigSpiChunkPayloadSize](#)
- Parameter [EthCtrlConfigSpiCommRetries](#)
- Parameter [EthCtrlConfigSpiEnableControlDataProtection](#)
- Parameter [EthCtrlConfigSpiEnableRxCSAlign](#)
- Parameter [EthCtrlConfigSpiEnableRxCutThrough](#)
- Parameter [EthCtrlConfigSpiEnableRxZeroAlign](#)
- Parameter [EthCtrlConfigSpiEnableTransmitDataHdrSequence](#)
- Parameter [EthCtrlConfigSpiEnableTxChecksum](#)
- Parameter [EthCtrlConfigSpiEnableTxCutThrough](#)
- Parameter [EthCtrlConfigSpiSelectTimeStamp](#)
- Parameter [EthCtrlConfigSpiTransmitCreditThreshold](#)
- Container [EthCtrlConfigSpiSequence](#)
- Parameter [EthCtrlConfigSpiAccessSynchronous](#)
- Reference [EthCtrlConfigSpiSequenceName](#)
- Container [CommonPublishedInformation](#)
 - * Parameter [ModuleId](#)
 - * Parameter [VendorId](#)
 - * Parameter [VendorApiInfix](#)
 - * Parameter [ArReleaseMajorVersion](#)
 - * Parameter [ArReleaseMinorVersion](#)

- * Parameter [ArReleaseRevisionVersion](#)
- * Parameter [SwMajorVersion](#)
- * Parameter [SwMinorVersion](#)
- * Parameter [SwPatchVersion](#)

4.1 Module Eth

Configuration of the Eth (Ethernet Driver) module.

Included containers:

- [EthGeneral](#)
- [EthConfigSet](#)
- [CommonPublishedInformation](#)

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-PRE-COMPILE, VARIANT-LINK-TIME, VARIANT-POST-BUILD

4.2 Container EthGeneral

General configuration of Ethernet Driver module.

Included subcontainers:

- [EthCtrlOffloading](#)
- [EthGeneralVendorSpecific](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.3 Parameter EthDevErrorDetect

Enables / Disables development error detection.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	true

4.4 Parameter EthGetDropCountApi

Obsolete and kept only for backwards compatibility. The Eth_GetDropCount API has been replaced with Eth_GetCounterVal since R4.3.1. Use parameter EthGetCounterValuesApi instead.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.5 Parameter EthGetEtherStatsApi

Obsolete and kept only for backwards compatibility. The Eth_GetEtherStats API has been replaced with the Eth_GetTxStats and Eth_GetRxStats APIs since R4.3.1.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.6 Parameter EthGetCounterValuesApi

Enables / Disables Eth_GetCounterValues API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.7 Parameter EthGetRxStatsApi

Enables / Disables Eth_GetRxStats API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.8 Parameter EthGetTxStatsApi

Enables / Disables Eth_GetTxStats API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.9 Parameter EthGetTxErrorCounterValuesApi

Enables / Disables Eth_GetTxErrorCounterValues API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.10 Parameter EthSendMultiBufferFrameApi

Enables / Disables Eth_SendMultiBufferFrame API.
available for controllers configured to use external buffers.

Important: This feature will only be

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.11 Parameter EthGlobalTimeSupport

Enables/Disables the GlobalTime APIs used amongst others by Global Time Synchronization over Ethernet.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false

Property	Value
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.12 Parameter EthVersionInfoApi

Enables / Disables Eth_GetVersionInfo API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.13 Parameter EthIndex

Specifies the InstanceId of this module instance. If only one instance is present it shall have the Id 0.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
default Value	0
max	255
min	0

4.14 Parameter EthMainFunctionPeriod

The period between successive calls to the main function in seconds.

Ethernet driver does not require this information but the BSW scheduler.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	0.001
max	65.535
min	0.001

4.15 Parameter EthMaxCtrlsSupported

Limits the total number of supported controllers.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE

Property	Value
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1
max	1
min	1

4.16 Parameter EthMaxBuffersInMultiBufferFrame

The maximum number of buffers that can be used in a multi buffer transmission.
This field is only used in the context of Eth_SendMultiBufferFrame API.

Note:

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	16
max	255
min	0

4.17 Reference EthEcucPartitionRef

Maps the Ethernet driver to zero or multiple ECUC partitions to make the module's API available in this partition.

The Ethernet driver will operate as an independent instance in each of the partitions.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
	VARIANT-POST-BUILD: PRE-COMPILE
multiplicityConfigClasses	

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
postBuildVariantValue	true
	VARIANT-POST-BUILD: PRE-COMPILE
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/EcuC/EcuPartitionCollection/EcuPartition

4.18 Container EthCtrlOffloading

Configuration of hardware offloading features.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.19 Parameter EthCtrlEnableOffloadChecksumICMP

Enables / Disables hardware offloading for ICMP checksums.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
	VARIANT-LINK-TIME: PRE-COMPILE
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.20 Parameter EthCtrlEnableOffloadChecksumIPv4

Enables / Disables hardware offloading for IPv4 checksums.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.21 Parameter EthCtrlEnableOffloadChecksumTCP

Enables / Disables hardware offloading for TCP checksums.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.22 Parameter EthCtrlEnableOffloadChecksumUDP

Enables / Disables hardware offloading for UDP checksums.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.23 Container EthGeneralVendorSpecific

General vendor-specific configuration of Ethernet Driver.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.24 Parameter EthDisableDemEventDetect

When this option is enabled, no DEM events are reported.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.25 Parameter EthEnableUserModeSupport

When this parameter is enabled, the Eth module will adapt to run from User Mode, with the following measures:

(if applicable) a) configuring REG_PROT for the Eth Controllers so that the registers under protection eth be accessed from user mode by setting UAA bit in REG_PROT_GCR to 1.

(if applicable) b) using 'call trusted function' stubs for all internal function calls that access registers requiring supervisor mode.

(if applicable) c) other module specific measures for more information, please see chapter 5.7 User Mode Support in IM

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.26 Parameter EthMulticoreSupport

Switches multicore support on or off:

False: For all variants, no EcucPartition shall be referenced in EthEcucPartitionRef.

True: For all variants, at least one EcucPartition needs to be referenced in EthEcucPartitionRef.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.27 Parameter EthUpdatePhysAddrFilterApi

Enables/Disables Eth_UpdatePhysAddrFilter API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.28 Parameter EthSwtManagementSupportApi

Enables / Disables the Switch Management APIs.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.29 Parameter EthCoalescingInterrupt

Enables / Disables Coalescing Interrupt.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.30 Parameter EthTimeoutMethod

Configures the timeout method.

Based on this selection a certain timeout method from OsIf will be used in the driver.

Note: If SystemTimer or CustomTimer are selected make sure the corresponding timer is enabled in OsIf General configuration.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	OSIF_COUNTER_DUMMY
literals	['OSIF_COUNTER_SYSTEM', 'OSIF_COUNTER_CUSTOM', 'OSIF_COUNTER_DUMMY']

4.31 Parameter EthTimeoutDuration

The unit of measurement is given in number of microseconds.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1000
max	65535
min	1

4.32 Parameter EthEnableCacheManagement

Enable/disable support for cache management inside the driver.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.33 Container EthConfigSet

This container contains the configuration parameters and sub containers of the AUTOSAR Eth module.

Included subcontainers:

- [EthCtrlConfig](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.34 Container EthCtrlConfig

Configuration of the individual controller

Included subcontainers:

- [EthCtrlVendorSpecific](#)
- [EthCtrlConfigEgress](#)
- [EthCtrlConfigTimeAwareShaper](#)

- [EthCtrlConfigIngress](#)
- [EthDemEventParameterRefs](#)
- [EthCtrlConfigSpiConfiguration](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	255
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.35 Parameter EthCtrlEnableMii

Enables / Disables Media Independent Interface (MII) for transceiver access.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.36 Parameter EthCtrlEnableMmd

Enables / Disables Clause 45 Media Independent Interface (MII) for transceiver access.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
default Value	false

4.37 Parameter EthCtrlEnableRxInterrupt

Enables / Disables receive interrupt.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	true

4.38 Parameter EthCtrlEnableTxInterrupt

Enables / Disables transmit interrupt.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.39 Parameter EthCtrlIdx

Specifies the instance ID of the configured controller.

Property	Value
type	ECUC-INTEGGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.40 Parameter EthCtrlMacLayerType

Defines the MAC layer type of the ethernet controller.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	ETH_MAC_LAYER_TYPE_XMII
literals	['ETH_MAC_LAYER_TYPE_XMII', 'ETH_MAC_LAYER_TYPE_XGMII', 'ETH_MAC_LAYER_TYPE_XXGMII']

4.41 Parameter EthCtrlMacLayerSubType

Defines the MAC layer subtype of the ethernet controller.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	STANDARD
literals	['REDUCED', 'SERIAL', 'STANDARD']

4.42 Parameter EthCtrlMacLayerSpeed

Defines the baud rate of the MAC layer.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true

Property	Value
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	ETH_MAC_LAYER_SPEED_10M
literals	['ETH_MAC_LAYER_SPEED_10M', 'ETH_MAC_LAYER_SPEED_100M', 'ETH_MAC_LAYER_SPEED_200M', 'ETH_MAC_LAYER_SPEED_1G', 'ETH_MAC_LAYER_SPEED_10G']

4.43 Parameter EthCtrlPhyAddress

Specifies the unique 48-bit physical address (MAC address) of the controller in network byte order.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	66:55:44:33:22:11

4.44 Parameter EthCtrlEnableSpiInterface

This optional parameter enables the processing of control data and over the SPI interface specific for MACPHY device. implies the respect of the SPI protocol

Ethernet frames
The use of this parameter
described in TC6 [26].

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC

Property	Value
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.45 Parameter EthCtrlConfigSwBufferHandling

Enables / Disables SW buffer management

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.46 Reference EthCtrlEcucPartitionRef

Maps the Ethernet controller to zero or one ECUC partition.

The ECUC partition referenced is a subset of the ECUC partitions where the Ethernet driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC

Property	Value
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/EcuC/EcuPartitionCollection/EcuPartition

4.47 Container EthCtrlVendorSpecific

Vendor specific configuration of the individual controller

Included subcontainers:

- [EthCtrlConfigGeneral](#)
- [EthCtrlConfigSafety](#)
- [EthCtrlConfigMac](#)
- [EthCtrlConfigPacketFilter](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.48 Container EthCtrlConfigGeneral

General Controller Configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.49 Parameter EthCtrlEnableAtInit

Enabled - The controller is enabled at initialization.

Disabled - The controller is left disabled at initialization and must be later enabled by calling Gmac_Ip_EnableController.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.50 Parameter EthCtrlAllocateTxDataBuffers

Enabled - The driver will allocate both Tx Buffer Descriptors and Tx Data Buffers.

Disabled - The driver will only allocate Tx Buffer Descriptors. The Tx Data Buffers will be allocated by the application.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.51 Parameter EthCtrlAllocateRxDataBuffers

Enabled - The driver will allocate both Rx Buffer Descriptors and Rx Data Buffers.

Disabled - The driver will only allocate Rx Buffer Descriptors. The Rx Data Buffers will be allocated by the application.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.52 Parameter EthCtrlEthIfIdx

Vendor specific: Defines the index of the controller in context of EthIf driver. This manual assignemnt of this value is only temporary workaround before proper solution is implemented.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	False
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	255
min	0

4.53 Parameter EthEthTrcvDriverVendorIdAndApiInfix

The vendor id and vendor api infix for the EthTrcv driver, if any.

This is used to include the correct header file and to point to the EthTrcv callbacks.

Example: 43_PHYGROUPB for a 43_PHYGROUPB driver or leave it blank to include EthTrcv.h

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	

4.54 Parameter EthEthSwtDriverVendorIdAndApiInfix

The vendor id and vendor api infix for the EthSwt driver, if any.

This is used to include the correct header file and to point to the EthSwt APIs.

Example: 43_SJA1110 for a 43_SJA1110 driver or leave it blank to include EthSwt.h

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	

4.55 Parameter EthDuplexMode

Defines the controller's duplex mode.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	ETH_FULL_DUPLEX
literals	['ETH_FULL_DUPLEX', 'ETH_HALF_DUPLEX']

4.56 Parameter EthTimeStampRequiredAccuracy

Value given in nanoseconds.

Represents the minimum required accuracy for this gPTP device.

To ensure there's enough headroom for clock drifts from both the GM and Slave, a maximum drift percentage per host must be assumed.

That is, $\text{TotalDriftPercentage} = \text{MaximumGmDriftPercentage} + \text{MaximumSlaveDriftPercentage}$. Then the following inequality must be true:

$$\text{EthTimeStampReferenceClock} * (1 + \text{TotalDriftPercentage} / 100) > (1 / \text{EthTimeStampRequiredAccuracy}) * 10^9.$$

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	40
max	4294967295
min	1

4.57 Reference EthModuleReferenceClock

Reference to the module clock for the GMAC.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Mcu/McuModuleConfiguration/McuClockSetting↔ Config/McuClockReferencePoint

4.58 Reference EthTimeStampReferenceClock

Reference to the Timestamp source clock for the GMAC.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/Mcu/McuModuleConfiguration/McuClockSetting↔ Config/McuClockReferencePoint

4.59 Container EthCtrlConfigSafety

Safety Controller Configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.60 Parameter EthCtrlSafetyCallback

Callback function invoked when a safety event is encountered.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	Eth_Ipw_SafetyIrqCallback

4.61 Parameter EthCtrlEnableEcc

Enables/Disables SECDDED data and address protection for MTL and EST memories.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.62 Parameter EthCtrlEnableDpp

Enables/Disables parity protection for Rx and Tx data paths.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.63 Parameter EthCtrlEnableFsmSpp

Enables/Disables state parity protection for the internal FSM.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.64 Parameter EthCtrlEnableFsmTimeout

Enables/Disables timeout protection for the internal FSM.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
	VARIANT-LINK-TIME: PRE-COMPILE
valueConfigClasses	

Property	Value
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.65 Reference ETH_E_SAFETY

Reference to the DemEventParameter which shall be issued when any safety error has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.66 Container EthCtrlConfigMac

MAC Controller Configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.67 Parameter MAC_CONFIG_CRC_STRIPPING

The last four bytes (FCS) of all packets of Ether type (type field greater than 1,536) are stripped and dropped before forwarding the packet to the application.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.68 Parameter MAC_CONFIG_AUTO_PAD

The MAC strips the Pad or FCS field on the incoming packets only if the value of the length field is less than 1,536 bytes.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.69 Parameter MAC_CONFIG_JUMBO_PKT_EN

The MAC allows jumbo packets of 9,018 bytes (9,022 bytes for VLAN tagged packets).

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.70 Parameter MAC_CONFIG_LOOPBACK

MAC operates in the loopback mode at GMII or MII. The Rx clock input (clk_rx_i) is required for the loopback to work properly.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.71 Parameter MAC_CONFIG_ENABLE_CRS_FD

The MAC transmitter checks the CRS signal before packet transmission in the full-duplex mode. The MAC starts the transmission only when the CRS signal is low.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.72 Parameter MAC_CONFIG_DISABLE_RECEIVE_OWN_HD

The MAC disables the reception of packets when the gmii_txen_o is asserted in the half-duplex mode.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.73 Parameter MAC_CONFIG_DISABLE_CRS_HD

The MAC transmitter ignores the (G)MII CRS signal during packet transmission in the half-duplex mode. As a result, no errors are generated because of Loss of Carrier or No Carrier during transmission.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.74 Parameter MAC_CONFIG_DISABLE_RETRY

The MAC attempts only one transmission. When a collision occurs on the (G)MII interface, the MAC ignores the current packet transmission and reports a Packet Abort with excessive collision error in the Tx packet status.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.75 Parameter MAC_CONFIG_DEFERRAL_CHECK_HD

The deferral check function is enabled in the MAC. The MAC issues a Packet Abort status, along with the excessive deferral error bit set in the Tx packet status, when the Tx state machine is deferred for more than 24,288 bit times in 10/100 Mbps mode (155,680 bits times in 1000/2500 Mbps operation). Deferral begins when the transmitter is ready to transmit, but it is prevented because of an active carrier sense signal (CRS) on (G)MII.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.76 Container EthCtrlConfigPacketFilter

Packet Filtering Configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.77 Parameter PKT_FILTER_RECV_ALL

Passes all received packets to the application, irrespective of whether they pass the address filter or not.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.78 Parameter PKT_FILTER_HASH_OR_PERFECT_FILTER

The address filter passes a packet if it matches either the perfect filtering or hash filtering.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	true

4.79 Parameter PKT_FILTER_BLOCK_CONTROL_PKTS

Blocks all control packets from reaching the application.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.80 Parameter PKT_FILTER_BLOCK_PAUSE_PKTS

Forwards all control packets except Pause packets to the application even if they fail the Address filter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.81 Parameter PKT_FILTER_PASS_ALL_CONTROL_PKTS

Forwards all control packets to the application even if they fail the Address filter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.82 Parameter

PKT_FILTER_PASS_CONTROL_PKTS_ADDR_MATCH

Forwards the control packets that pass the Address filter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.83 Parameter PKT_FILTER_DISABLE_BROADCAST

Blocks all incoming broadcast packets.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.84 Parameter PKT_FILTER_PASS_ALL_MULTICAST

All received packets with a multicast destination address are passed.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.85 Parameter PKT_FILTER_DST_ADDR_INV_FILTER_EN

The Address checking operates in inverse filtering mode for the destination address comparison for both unicast and multicast packets.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.86 Parameter PKT_FILTER_HASH_MULTICAST

The destination address filtering of received multicast packets is done according to the hash table.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.87 Parameter PKT_FILTER_HASH_UNICAST

The destination address filtering of received unicast packets is done according to the hash table.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.88 Parameter PKT_FILTER_PROMISCUOUS_MODE

Passes all incoming packets irrespective of the destination or source address.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
default Value	false

4.89 Container EthCtrlConfigEgress

Controller Configuration Egress: Configuration of one Ethernet controller egress behavior.

Included subcontainers:

- [EthCtrlConfigEgressFifo](#)
- [EthCtrlConfigScheduler](#)
- [EthCtrlConfigShaper](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.90 Reference EthCtrlConfigEgressLastSchedulerRef

Reference to the scheduler which is the last in the Egress structure.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfigEgress/EthCtrlConfigScheduler

4.91 Container EthCtrlConfigEgressFifo

Controller Configuration Egress: Represents a FIFO at the egress side.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.92 Parameter EthCtrlConfigEgressFifoBufLenByte

Length of FIFO elements in bytes.

Note that this represents the length of a single packet buffer (including the 14-bytes Ethernet frame header and the 4-bytes FCS).

Therefore, the payload length will be equal to EthCtrlConfigEgressFifoBufLenByte - 18.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	64
max	1536
min	64

4.93 Parameter EthCtrlConfigEgressFifoBufTotal

FIFO Buffer Count (defines the number of packet buffers of length EthCtrlConfigEgressFifoBufLenByte).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	16
max	256
min	4

4.94 Parameter EthCtrlConfigEgressFifoIdx

Egress FIFO index.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	0
max	2
min	0

4.95 Parameter EthCtrlConfigEgressFifoCallback

Callback function invoked when an Egress (Tx) FIFO event is encountered.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	Eth_TxIrqCallback

4.96 Parameter EthCtrlConfigEgressFifoPriorityAssignment

Message egress priority assignment.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	7
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.97 Container EthCtrlConfigScheduler

Controller Configuration Scheduler: Represents a Scheduler on the Egress side.

Included subcontainers:

- [EthCtrlConfigSchedulerPredecessor](#)
- [EthCtrlConfigSchedulerVendorSpecific](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	255
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.98 Container EthCtrlConfigSchedulerPredecessor

Controller Configuration Scheduler Predecessors: Defines an ordered list of predecessors for this scheduler.

Included subcontainers:

- [EthCtrlConfigSchedulerPredecessorVendorSpecific](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.99 Parameter EthCtrlConfigSchedulerPredecessorOrder

Defines the order of the scheduler predecessors.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	255
min	0

4.100 Reference EthCtrlConfigSchedulerPredecessorRef

Choice reference to the scheduler predecessor.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/AUTOSAR/EcuDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfig↔Egress/EthCtrlConfigEgressFifo', '/AUTOSAR/EcuDefs/Eth/EthConfigSet/↔EthCtrlConfig/EthCtrlConfigEgress/EthCtrlConfigScheduler', '/AUTOSA↔R/EcuDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfigEgress/EthCtrl↔ConfigShaper']

4.101 Container EthCtrlConfigSchedulerPredecessorVendorSpecific

Scheduler Predecessor vendor specific configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.102 Parameter EthQueueBandwidth

Represents the bandwidth percentage allocated for this queue.

In SP, this parameter is irrelevant.

In WRR, $\text{EthQueueBandwidth} / 100 * \text{EthTotalPacketsPerCycle}$ must not exceed 100 packets.

In DWRR, $\text{EthQueueBandwidth} / 100 * \text{EthTotalQuantumPerCycle}$ must not exceed 1,250,000 bytes.

This field is editable only when the scheduling algorithm is set to anything other than STRICT_PRIORITY.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	100
max	100
min	1

4.103 Container EthCtrlConfigSchedulerVendorSpecific

Scheduler vendor specific configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.104 Parameter EthTxSchedulerAlgorithm

Configures the global Tx Scheduling Algorithm affecting all enabled Tx Queues.

For both round-robin algorithms, the queues are serviced in $Q_n > Q_{n-1} > \dots > Q_0$ order based on their allocated bandwidths.

When Credit-Based Shapers are configured in EthCtrlConfigSchedulerPredecessor, the scheduling algorithm must be set to STRICT_PRIORITY.

In a single Egress FIFO configuration, the scheduling algorithm must be set to STRICT_PRIORITY.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	STRICT_PRIORITY
literals	['STRICT_PRIORITY', 'WEIGHTED_ROUND_ROBIN', 'WEIGHTED_FAIR_QUEUEING', 'DEFICIT_WEIGHTED_ROUND_ROBIN']

4.105 Parameter EthTotalPacketsPerCycle

Configures the total weight value (in packets) per arbitration cycle from which individual queue weights are calculated based on their allocated bandwidths.

For example, if queues (Q2, Q1, Q0) are configured with bandwidths (50%, 25%, 25%) and EthTotalPacketsPerCycle = 4 packets,

then the individual queue weights are $(4 * 0.5, 4 * 0.25, 4 * 0.25) = (2, 1, 1)$ packets.

This field is editable only when the scheduling algorithm is set to WEIGHTED_ROUND_ROBIN.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	10
max	50000000
min	1

4.106 Parameter EthTotalQuantumPerCycle

Configures the total quantum value (in bytes) per arbitration cycle from which individual queue quantum values are calculated based on their allocated bandwidths.

For example, if queues (Q2, Q1, Q0) are configured with bandwidths (50%, 25%, 25%) and EthTotalQuantumPerCycle = 2000 bytes,

then the individual queue quantum values are $(2000 * 0.5, 2000 * 0.25, 2000 * 0.25) = (1000, 500, 500)$ bytes.

This field is editable only when the scheduling algorithm is set to DEFICIT_WEIGHTED_ROUND_ROBIN.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1000
max	4294967295
min	1

4.107 Container EthCtrlConfigShaper

Controller Configuration Shaper : Represents a Shaper at the Egress side.

Included subcontainers:

- [EthCtrlConfigShaperVendorSpecific](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.108 Parameter EthCtrlConfigShaperIdleSlope

Defines the increase of credit in bits per second for the AVB shaper.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	75000000
min	0

4.109 Parameter EthCtrlConfigShaperMaxCredit

Maximum amount of credits that can be accumulated for a queue.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	49
min	0

4.110 Parameter EthCtrlConfigShaperMinCredit

Minimum amount of credits in bytes that can be accumulated for a queue.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	9
min	0

4.111 Reference EthCtrlConfigShaperPredecessorFifoRef

Reference to the FIFO which is the predecessor for this Shaper.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfig← Egress/EthCtrlConfigEgressFifo

4.112 Container EthCtrlConfigShaperVendorSpecific

Shaper vendor specific configuration.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.113 Parameter EthCtrlConfigHiCredit

Defines the maximum value (in bits) that can be accumulated in the credit parameter.

This value should be equal to the maximum size (in bits) of an interfering (best-effort) frame.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	12000
max	131072
min	0

4.114 Parameter EthCtrlConfigLoCredit

Defines the minimum value (in bits) that can be accumulated in the credit parameter.

This value should be equal to twice the maximum size (in bits) of a frame transmitted from this queue.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	4096
max	131072
min	0

4.115 Container EthCtrlConfigTimeAwareShaper

Controller Time Aware Shaper: Represents Time Aware Shaper configuration on the Egress side.

Included subcontainers:

- [EthCtrlConfigTimeAwareShaper](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.116 Container EthCtrlConfigTimeAwareShaper

Controller Time Aware Shaper: Represents Time Aware Shaper configuration on the Egress side.

Included subcontainers:

- [EthGateControlList](#)
- [EthPreemptionClassification](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.117 Parameter EthGateConfigBaseTimeInSecond

The start time for the Gate control list

This value in second.

Once the execution of a given list begins, the implementation can update this value to indicate the next list execution begin time.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1000
max	4294967295
min	0

4.118 Parameter EthGateConfigBaseTimeInNanoSecond

The start time for the Gate control list

This value in nanosecond.

Once the execution of a given list begins, the implementation can update this value to indicate the next list execution begin time.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1000
max	4294967295
min	0

4.119 Parameter EthGateConfigCycleTime

The time at which the execution of the gate control list should be repeated.

This value in nanosecond.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	259294967295
min	0

4.120 Parameter EthGateConfigExtendTime

The parameter holds the amount of time (in nanoseconds) can be extended before switching to the new Gate Control List.

This will be useful in avoiding small fragments of the current list being executed before switching to a new list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	2147483647
min	0

4.121 Parameter EthCtrlEnablePreemption

Enabled - Frame preemption is enabled.

Disabled - Frame preemption is disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.122 Parameter EthReleaseAdvanceTime

The maximum time in nanoseconds elapse between issuing a RELEASE to the MAC and the MAC being ready to resume

transmission of preemptable frames.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	1000
max	65535
min	1

4.123 Parameter EthHoldAdvanceTime

The maximum time in nanoseconds elapse between issuing a HOLD to the MAC and the MAC ceasing to transmit any preemptable frame.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1000
max	65535
min	1

4.124 Container EthGateControlList

Gate Control List : governs the gate close or open events based on schedule provided.

Included subcontainers:

- [EthGateControlFifo](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.125 Parameter EthCtrlConfigInterval

The time in nano seconds for gate controls are valid and should be applied before reading the next

gate controls from the list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	16000000
min	0

4.126 Container EthGateControlFifo

Gate control fifo: Configure Fifo is open or close in duration interval.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.127 Parameter EthCtrlConfigGateStatus

This parameters to select gate status is open or close.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OPEN
literals	['CLOSE', 'OPEN']

4.128 Reference EthCtrlConfigFifoRef

Reference to the FIFO which is the predecessor for this time aware shaper.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfig↔Egress/EthCtrlConfigEgressFifo

4.129 Container EthPreemptionClassification

Preemption classification : select reference fifo is preemptable or express.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.130 Parameter EthCtrlConfigPreemptionClassification

This parameters to select the queue is preemptable or express

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	EXPRESS
literals	['PREEMPTABLE', 'EXPRESS']

4.131 Reference EthPreemptionFifoRef

Reference to the FIFO which is the predecessor for preemption.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false

Property	Value
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Eth/EthConfigSet/EthCtrlConfig/EthCtrlConfig↔ Egress/EthCtrlConfigEgressFifo

4.132 Container EthCtrlConfigIngress

Controller Configuration Ingress: Configuration of one Ethernet controller ingress behavior.

Included subcontainers:

- [EthCtrlConfigIngressFifo](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.133 Container EthCtrlConfigIngressFifo

Controller Configuration Ingress : Represents a FIFO at the ingress side.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.134 Parameter EthCtrlConfigIngressFifoBufLenByte

Length of FIFO elements in bytes.

Note that this represents the length of a single packet buffer (including the 14-bytes Ethernet frame header and the 4-bytes FCS).

Therefore, the payload length will be equal to EthCtrlConfigIngressFifoBufLenByte - 18.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	64
max	1536
min	64

4.135 Parameter EthCtrlConfigIngressFifoBufTotal

FIFO Buffer Count (defines the number of packet buffers of length EthCtrlConfigIngressFifoBufLenByte).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	16
max	256
min	4

4.136 Parameter EthCtrlConfigIngressFifoIdx

Ingress FIFO index.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	0
max	2
min	0

4.137 Parameter EthCtrlConfigIngressFifoCallback

Callback function invoked when an Ingress (Rx) FIFO event is encountered.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	Eth_RxIrqCallback

4.138 Parameter EthCtrlConfigIngressFifoPriorityAssignment

Message ingress priority assignment.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	7
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.139 Container EthDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the API Dem_ReportErrorStatus in case the corresponding error occurs.

The EventId is taken from the referenced DemEventParameter's DemEventId value.

The standardized errors are provided in the container and can be extended by vendor specific error references.

Container is editable only when EthGeneral/EthGeneralVendorSpecific/EthDisableDemEventDetect = FALSE.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.140 Reference ETH_E_ACCESS

Reference to the DemEventParameter which shall be issued when the error 'Controller access failed' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.141 Reference ETH_E_RX_FRAMES_LOST

Reference to the DemEventParameter which shall be issued when the error 'Rx Frame Lost' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.142 Reference ETH_E_CRC

Reference to the DemEventParameter which shall be issued when the error 'CRC Failure' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.143 Reference ETH_E_UNDERSIZEFRAME

Reference to the DemEventParameter which shall be issued when the error 'Undersized Frame' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.144 Reference ETH_E_OVERSIZEFRAME

Reference to the DemEventParameter which shall be issued when the error 'Oversized Frame' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.145 Reference ETH_E_ALIGNMENT

Reference to the DemEventParameter which shall be issued when the error 'Alignment Error' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.146 Reference ETH_E_SINGLECOLLISION

Reference to the DemEventParameter which shall be issued when the error 'Single Collision' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.147 Reference ETH_E_MULTIPLECOLLISION

Reference to the DemEventParameter which shall be issued when the error 'Multiple Collisions' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.148 Reference ETH_E_LATECOLLISION

Reference to the DemEventParameter which shall be issued when the error 'Late Collisions' has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.149 Container EthCtrlConfigSpiConfiguration

SPI Interface configuration of one Ethernet controller (MACPHY use).

Included subcontainers:

- [EthCtrlConfigSpiSequence](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.150 Parameter EthCtrlConfigSpiCommTimeout

Indicates the maximum time allowed to the Ethernet controller for

replying (either positively or negatively) to a SPI command. Timeout is configured in seconds. Timeout value of 0 means that no specific timeout is to be used by Ethernet controller and the communication is executed at the best of the SPI HW capacity

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.0
max	0.1
min	0.0

4.151 Parameter EthCtrlConfigSpiChunkPayloadSize

Configures the size of the payload chunks which will be transferred over the SPI interface. Note: The chunk is the basic element for data transaction over the SPI which can be a section of an Ethernet frame or management command. The configured value has to be a multiple of 8.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	64
max	64
min	8

4.152 Parameter EthCtrlConfigSpiCommRetries

Indicates the maximum number of communication retries in case of a failed SPI communication (applies both to timed out communication and to errors/NACK in the response data). If configured value is 0, no retry is allowed (communication is expected to succeed at first try).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	254
min	0

4.153 Parameter EthCtrlConfigSpiEnableControlDataProtection

Enables the control data protection. When set, all control data written to and read from the MACPHY will be transferred with its complement for detection of bit errors as defined in OA TC6 [26]. FALSE: Control data read/write protection is disabled (unprotected). TRUE: Control data read/write protection is enabled (protected).

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.154 Parameter EthCtrlConfigSpiEnableRxCSAlign

Configures the CSn Align Receive frame. TRUE: all received Ethernet

frames data shall start at the beginning of the first receive data chunk

payload following CSn assertion FALSE: received frames may begin

within any receive data chunk of the transaction when this bit is clear.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.155 Parameter EthCtrlConfigSpiEnableRxCutThrough

When supported by the HW, enables the cut through mode of frame



Tresos Configuration Plug-in

from the network to the SPI host.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.156 Parameter EthCtrlConfigSpiEnableRxZeroAlign

Configures the zero-align receive frame. TRUE: all received Ethernet frames data shall be aligned to start at the beginning of any receive data chunk payload. FALSE: Received frames may begin anywhere within the receive data chunk payload

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.157 Parameter EthCtrlConfigSpiEnableTransmitDataHdrSequence

When supported by the HW, enables the transmit data sequence

monitoring. FALSE: transmit data header sequence bit monitoring

disabled. TRUE: transmit data header sequence bit monitoring

enabled

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
default Value	false

4.158 Parameter EthCtrlConfigSpiEnableTxChecksum

Configures the CSn Align Receive frame. TRUE: all received Ethernet

frames data shall start at the beginning of the first receive data chunk

payload following CSn assertion FALSE: received frames may begin

within any receive data chunk of the transaction when this bit is clear

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK

Property	Value
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.159 Parameter EthCtrlConfigSpiEnableTxCutThrough

Configures the CSn Align Receive frame. TRUE: all received Ethernet frames data shall start at the beginning of the first receive data chunk payload following CSn assertion FALSE: received frames may begin within any receive data chunk of the transaction when this bit is clear

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.160 Parameter EthCtrlConfigSpiSelectTimeStamp

When timestamp supported by the HW, selects size and format of the timestamps. FALSE: 32-bits timestamps TRUE: 64-bit timestamps

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.161 Parameter EthCtrlConfigSpiTransmitCreditThreshold

Configures the minimum of available transmit credit before the writing

IRQn is asserted. As per OA TC6, this information is notified by the

TXC field. 0 = 1 credit 1 = 4 credits 2 = 8 credits 3 = 16 credits

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	3
min	0

4.162 Container EthCtrlConfigSpiSequence

Container gives Ethernet controller driver information about one

SPI sequence. One SPI sequence used by Ethernet controller driver is in exclusive use for it. No other driver is allowed to access this sequence. Ethernet controller driver may use one sequence to access n Ethernet controller hardware chips of the same type or n sequences are used to access one single Ethernet controller hardware chip. If a Ethernet controller hardware has no SPI interface, there is no instance of this container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.163 Parameter EthCtrlConfigSpiAccessSynchronous

This parameter is used to define whether the access to the Spi sequence is synchronous or asynchronous. true: SPI access is synchronous. false: SPI access is asynchronous

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.164 Reference EthCtrlConfigSpiSequenceName

Reference to a Spi sequence configuration container.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiSequence

4.165 Container CommonPublishedInformation

Common container, aggregated by all modules. It contains published information about vendor and versions.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.166 Parameter ModuleId

Module ID of this module from the BSW Module List.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	88
max	88
min	88

4.167 Parameter VendorId

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	43
max	43
min	43

4.168 Parameter VendorApiInfix

In driver modules which can be instantiated several times on a single ECU,

BSW00347 requires that the name of APIs is extended by the VendorId and a vendor specific name.

This parameter is used to specify the vendor specific name.

In total, the implementation specific name is generated as follows:

[ModuleName]_[VendorId]_[VendorApiInfix][API name from SWS].

E.g. Assuming that the VendorId of the implementor is 123 and the implementer chose a

VendorApiInfix of 'v11r456' an API named Can_Write defined in the SWS will translate to Can_123_v11r456Write.

This parameter is mandatory for all modules with upper multiplicity > 1.

It shall not be used for modules with upper multiplicity = 1.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	

4.169 Parameter ArReleaseMajorVersion

Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	4
max	4
min	4

4.170 Parameter ArReleaseMinorVersion

Minor version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	7
max	7
min	7

4.171 Parameter ArReleaseRevisionVersion

Revision version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.172 Parameter SwMajorVersion

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	3
max	3
min	3

4.173 Parameter SwMinorVersion

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.174 Parameter SwPatchVersion

Patch version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0



Chapter 5

Module Index

5.1 Software Specification

Here is a list of all modules:

ETH Driver	122
GMAC Driver	123
Ethernet Driver	177



Chapter 6

Data Structure Index

6.1 Data Structures

Here are the data structures with brief descriptions:

Gmac_Ip_ManagementInfo	
Management Frame information	184



Chapter 7

Module Documentation

7.1 ETH Driver

7.1.0.1 Functions Expectation called for transmission in polling mode:

- Eth_ProvideTxBuffer
- Eth_Transmit
- Eth_TxConfirmation

7.2 GMAC Driver

7.2.1 Detailed Description

7.2.1.1 Platform-specific prerequisites

It is the user's responsibility to configure `DCM_GPR_DCMRWF1[EMAC_CONF_SEL]` to the desired PHY interface. It shall match the interface expected by the PHY and the MAC.

7.2.1.2 How to use the GMAC driver in your application

The GMAC driver does not handle clock setup or any kind of pin configuration. This is handled by the **Clock Driver** and **Pin Driver** modules, respectively. The driver assumes that the correct clock and pins configurations have been made, so it is the user's responsibility to set up clocking and pin configurations correctly.

In order to use the GMAC driver in your application, the [Gmac_Ip_Init\(\)](#) function should be called prior to using the rest of the API. The parameters of this function specify:

- The GMAC instance to be initialized
- A structure which will hold the internal state of the driver
- A structure specifying the configuration of the GMAC module
- Two arrays of structures specifying the configuration of the buffers (one structure for each rings - an array for Rx and one for Tx)
- The MAC address to be configured for the module

The configuration of the module is specified through the `Gmac_Ip_ConfigType` structure and contains:

- MII-related configurations (mode, speed, duplex)
- Configuration options for the receive and transmit path
- The maximum frame length
- The MAC interrupt sources which should be enabled
- A callback function to be invoked on events
- The number of Rx and Tx rings used by the driver
- MAC address filtering options

The buffers configuration is specified through the [Gmac_Ip_TxRingConfigType](#) and [Gmac_Ip_RxRingConfigType](#) structures and contains:

- The size of the ring
- Pointers to the beginning of the buffer descriptor rings

- Pointers to the beginning of the memory area where the buffer data resides
- Channel interrupt sources
- A callback function to be invoked on per channel events
- The DMA burst length
- A list of VLAN priorities
- The queue operation mode - only for Tx rings
- The scheduling weight/quantum for DCB queues - only for Tx rings
- Idle and send slope credits for AVB queues - only for Tx rings
- High and low credit limits for AVB queues - only for Tx rings

All untagged packets will be received by ring 0 (zero). VLAN tagged packets are routed depending on the VLAN Tag Priority field and the way these priorities were assigned to the rings in the configuration.

In order to de-initialize the driver, the [Gmac_Ip_Deinit\(\)](#) function shall be used. This function will disable the GMAC interrupts and the module, so calling other GMAC driver functions after de-initializing the driver will have undefined behavior. In order to use the driver again, [Gmac_Ip_Init\(\)](#) should be called.

7.2.1.3 How to enable zero-copy optimization

The buffers allocation is controlled by the following configuration parameters:

- EthCtrlAllocateTxDataBuffers
- EthCtrlAllocateRxDataBuffers

Setting them both to FALSE will prevent the driver from using internal buffers.

However, it is important to keep in mind that the programming sequences will also be slightly affected:

- EthAllocateTxDataBuffers = TRUE
 - Call sequence for each frame: Gmac_Ip_GetTxBuff -> Gmac_Ip_SendFrame
 - Gmac_Ip_SendFrame must be called with the same buffer returned by Gmac_Ip_GetTxBuff
- EthAllocateTxDataBuffers = FALSE
 - Call sequence for each frame: Gmac_Ip_SendFrame
 - Gmac_Ip_SendFrame can be called with any buffer provided by the application
- EthCtrlAllocateRxDataBuffers = TRUE
 - Call sequence for each frame: Gmac_Ip_ReadFrame -> Gmac_Ip_ProvideRxBuff
 - Gmac_Ip_ProvideRxBuff must be called with the same buffer returned by Gmac_Ip_ReadFrame

- EthCtrlAllocateRxDataBuffers = FALSE

1. Application has an RX buffer pool

- After controller initialization: Call Gmac_Ip_ProvideRxBuff for each buffer
- Call sequence for each frame: Gmac_Ip_ReadFrame -> Gmac_Ip_ProvideRxBuff
- Gmac_Ip_ProvideRxBuff can be called with either:
 - * The same buffer returned by Gmac_Ip_ReadFrame if the application wants to reuse the buffer
 - * A buffer different than the one returned by Gmac_Ip_ReadFrame if the application doesn't want to reuse the buffer

2. Application doesn't have an RX buffer pool and wants to provide RX buffers on demand

- After controller initialization: Nothing to be done
- Call sequence for each frame: Gmac_Ip_ProvideRxBuff -> Gmac_Ip_ReadFrame

7.2.1.4 How to calculate size of ethernet buffer areas

The contents of this section are relevant only when EthCtrlAllocateTxDataBuffers = TRUE and/or EthCtrlAllocateRxDataBuffers = TRUE.

Ethernet buffer areas is placed in non-cacheable memory for each queue in each controller with order:

- Rx descriptor buffer areas: number of buffers configured (by EthCtrlConfigIngressFifoBufTotal) * size of each buffer (Gmac_Ip_BufferDescriptorType).
- Rx data buffer areas: number of buffers configured (by EthCtrlConfigIngressFifoBufTotal) * size of each buffer (EthCtrlConfigIngressFifoBufLenByte).
- Tx descriptor buffer areas: number of buffers configured (by EthCtrlConfigEgressFifoBufTotal) * size of each buffer (Gmac_Ip_BufferDescriptorType).
- Tx data buffer areas: number of buffers configured (by EthCtrlConfigEgressFifoBufTotal) * size of each buffer (EthCtrlConfigEgressFifoBufLenByte).

All of above memory areas need to align with start address by FEATURE_GMAC_BUFFDESCR_ALIGNMENTS_BYTES(8U).

This configuration will be placed in loop with multiple queues and multiple controllers.

7.2.1.4.1 In code: - For Reception buffer descriptors for Rx Ring ["num:i(\$RingIdx)"] :

```
VAR_ALIGN(Gmac_Ip_BufferDescriptorType GMAC_["num:i($CtrlIdx)"]_RxRing_["num:i($RingIdx)"]_DescBuffer[GMA
FEATURE_GMAC_BUFFDESCR_ALIGNMENT_BYTES)
```

- For Reception data buffers for Rx Ring ["num:i(\$RingIdx)"]

```
VAR_ALIGN(uint8 GMAC_["num:i($CtrlIdx)"]_RxRing_["num:i($RingIdx)"]_DataBuffer[GMAC_["num:i($CtrlIdx)"]_M
* GMAC_["num:i($CtrlIdx)"]_MAX_RXBUFFLEN_SUPPORTED], FEATURE_GMAC_BUFFDATA_ALIGNMENT_BY
```

- For Transmission buffer descriptors for Tx Ring ["num:i(\$RingIdx)"]

```
VAR_ALIGN(Gmac_Ip_BufferDescriptorType GMAC_["num:i($CtrlIdx)"]_TxRing_["num:i($RingIdx)"]_DescBuffer[GMA
FEATURE_GMAC_BUFFDESCR_ALIGNMENT_BYTES)
```

- For Transmission data buffers for Tx Ring ["num:i(\$RingIdx)"]

```
VAR_ALIGN(uint8 GMAC_["num:i($CtrlIdx)"]_TxRing_["num:i($RingIdx)"]_DataBuffer[GMAC_["num:i($CtrlIdx)"]_M
* GMAC_["num:i($CtrlIdx)"]_MAX_TXBUFFLEN_SUPPORTED], FEATURE_GMAC_BUFFDATA_ALIGNMENT_BY
```

7.2.1.5 How to set up MAC address filters

The Address Filtering Module of the MAC checks the source address and destination address fields of each incoming packet. By default, the module only receives the frames having the destination address equal to the MAC address with which the GMAC module has been initialized or the broadcast address.

The module has 1 additional 48-bit perfect address filter with masks for each byte, which can be configured for either source address or destination address filtering. This filter can be set-up by calling the [Gmac_Ip_SetAddrPerfectFilter\(\)](#) function.

Multicast and unicast destination addresses can be added to the hash filter by calling the [Gmac_Ip_AddDstAddrToHashFilter\(\)](#) function and removed by calling the [Gmac_Ip_RemoveDstAddrFromHashFilter\(\)](#) function.

The choice between perfect and hash filters can be done by calling the [Gmac_Ip_SetUnicastHashFilter\(\)](#), [Gmac_Ip_SetMulticastHashFilter\(\)](#) and [Gmac_Ip_SetHashOrPerfectFilter\(\)](#) functions.

For multicast and broadcast frames, the filters can be bypassed by calling the [Gmac_Ip_SetBroadcastForwardAll\(\)](#) and [Gmac_Ip_SetMulticastForwardAll\(\)](#) functions.

7.2.1.6 How to set up VLAN control and filters

Processing of IEEE802.1Q VLAN tags in both transmitted and received packets can be enabled and configured by calling the [Gmac_Ip_EnableVlan\(\)](#) function. This function takes a parameter of type [Gmac_Ip_VlanConfigType](#) which encapsulates the following properties:

- Enables processing of up to two VLAN tags in transmitted and received packets (if present)
- Enables the transmitter and receiver to consider the S-VLAN packets (Type = 0x88A8) as valid VLAN tagged packets
- Selects if outer VLAN tag and type are deleted, inserted or replaced in transmitted packets

- Selects if inner VLAN tag and type are deleted, inserted or replaced in transmitted packets; effective only if double VLAN processing is enabled
- Selects if outer VLAN tag is stripped in received packets
- Selects if inner VLAN tag is stripped in received packets

The user has the option to configure both outer VLAN tag and type for an active transmission ring, to be automatically inserted or replaced for each packet, by calling the [Gmac_Ip_SetTxOuterVlanTagForInsertion\(\)](#) and [Gmac_Ip_SetTxOuterVlanTagForReplacement](#) functions. Additionally, both inner VLAN tag and type can be configured for all transmission rings, by calling the [Gmac_Ip_SetTxInnerVlanTag\(\)](#) function.

By default, all VLAN tagged packets are accepted. The Rx filter can be enabled and configured by calling the [Gmac_Ip_SetVlanTagRxFilter\(\)](#) function. The matching options that can be configured are the following:

- Enable matching on the inner VLAN tag (if present); by default, the outer VLAN tag is compared
- Enable matching for S-VLAN (Type = 0x88A8) packets; by default, matching is performed for C-VLAN (Type = 0x8100) packets
- Disable matching for the VLAN type (only the VLAN tag is compared); by default, the VLAN type is compared
- Enable VLAN tag inverse filter match
- Enable 12-bit VLAN identifier match; by default, the complete 16-bit VLAN tag is compared

The available perfect filters (see the `FEATURE_GMAC_VLAN_RX_FILTERS_NUM` define for the the specific maximum number of filters configurable on each platform) can be set, read and cleared by the means of the following functions: [Gmac_Ip_AddVlanTagToRxFilter\(\)](#), [Gmac_Ip_ReadVlanTagRxFilter\(\)](#), [Gmac_Ip_ClearVlanTagRxFilter\(\)](#). These filters are used for perfect matching and the above described matching criteria are applied to all of them, based on the user configuration. Optionally, a hash table can be used for group matching of 16-bit outer VLAN tags, irrespective of the VLAN type or other criteria. VLAN tags can be added or removed from to/from the hash table by calling the [Gmac_Ip_AddVlanTagToHashTable\(\)](#) or [Gmac_Ip_RemoveVlanTagFromHashTable\(\)](#) functions.

When the Rx filters are enabled, the packets with unmatching VLAN tags will be rejected.

7.2.1.7 How to set up the System Time and enable timestamping for packets

For the timestamp to be captured for Tx and Rx packets, the System Time must be first initialized by calling the [Gmac_Ip_InitSysTime\(\)](#) function. The timestamp is available as extended information in [Gmac_Ip_TxInfoType](#) and [Gmac_Ip_RxInfoType](#) structures, passed as parameters to [Gmac_Ip_GetTransmitStatus\(\)](#) and [Gmac_Ip_ReadFrame\(\)](#).

Please note that the clock source for the GMAC System Time is not handled by the GMAC driver and the user must ensure that it is correctly enabled prior to initializing the System Time.

7.2.1.8 Examples:

7.2.1.8.1 Initializing the module: [Gmac_Ip_Init](#)(INST_GMAC_0, &Gmac_0_ConfigPB_<functionalGroupName>);

```
/* ... */
```

```
Gmac\_Ip\_Deinit(INST_GMAC_0);
```

7.2.1.8.2 Sending a frame (EthCtrlAllocateTxDataBuffers = TRUE): `Gmac_Ip_BufferType` buffer;

`Gmac_Ip_TxOptionsType` options = {FALSE, `GMAC_CRC_AND_PAD_INSERTION`, `GMAC_CHECKSUM_INSERTION`};

buffer.length = 8;

if (`GMAC_STATUS_SUCCESS` == `Gmac_Ip_GetTxBuff`(INST_GMAC_0, 0, &buffer, NULL_PTR))

```
{
    for (uint32 i = 0; i < 8; ++i)
    {
        buffer.data[i] = i;
    }
    Gmac_Ip_SendFrame(INST_GMAC_0, 0, &buffer, &options);
}
```

7.2.1.8.3 Sending a frame (EthCtrlAllocateTxDataBuffers = FALSE): `Gmac_Ip_BufferType` buffer;

`Gmac_Ip_TxOptionsType` options = {FALSE, `GMAC_CRC_AND_PAD_INSERTION`, `GMAC_CHECKSUM_INSERTION`};

uint8_t data[8] = {0, 1, 2, 3, 4, 5, 6, 7};

buffer.data = data;

buffer.length = 8;

`Gmac_Ip_SendFrame`(INST_GMAC_0, 0, &buffer, &options);

7.2.1.8.4 Receiving a frame - polling method (EthCtrlEnableRxInterrupt = FALSE): `Gmac_Ip_BufferType` buffer;

`Gmac_Ip_RxInfoType` info;

for (;;)

```
{
    if (status == Gmac_Ip_ReadFrame(INST_GMAC_0, 0, &buffer, &info))
    {
        /* Process buffer */

        /* buffer is no longer needed, provide it to the driver in order to be used by the reception mechanism */
        Gmac_Ip_ProvideRxBuff(INST_GMAC_0, 0, &buffer);
    }
}
```

7.2.1.8.5 Receiving a frame - interrupt method (EthCtrlEnableRxInterrupt = TRUE): void rx_callback(uint8 instance, uint8 channel)

```
{
    Gmac_Ip_BufferType buffer;

    if (status == Gmac_Ip_ReadFrame(INST_GMAC_0, 0, &buffer, NULL_PTR))
    {
        /* Process buffer */

        /* buffer is no longer needed, provide it to the driver in order to be used by the reception mechanism */
        Gmac_Ip_ProvideRxBuff(INST_GMAC_0, 0, &buffer);
    }
}

int main(void)
{
    /* ... */

    Gmac_Ip_Init(INST_GMAC_0, &Gmac_0_ConfigPB_<functionalGroupName>);

    /* ... */
}
```

7.2.1.8.6 Enabling and configuring VLAN processing and filtering: Gmac_Ip_VlanConfigType gmac_0_vlanConfig = {

```
    /* .enDoubleVlan = */ TRUE,

    /* .enSvlan = */ FALSE,

    /* .outerVlanIns = */ GMAC_VLAN_TAG_INSERTION,

    /* .innerVlanIns = */ GMAC_VLAN_TAG_REPLACEMENT,

    /* .outerVlanStrip = */ GMAC_VLAN_TAG_DO_NOT_STRIP,

    /* .innerVlanStrip = */ GMAC_VLAN_TAG_STRIP_IF_FILTER_FAIL,
};

Gmac_Ip_VlanRxFilterType gmac_0_vlanRxFilterConfig = {
```

```
/* .enInnerVlanMatch = */ FALSE,

/* .enSvlanMatch = */ FALSE,

/* .disVlanTypeMatch = */ FALSE,

/* .enInverseMatch = */ FALSE,

/* .en12bitMatch = */ TRUE,

};

/* Enable VLAN tag processing. */

Gmac_Ip_EnableVlan(INST_GMAC_0, &gmac_0_vlanConfig);

/* Set the outer VLAN tag to inserted for each transmitted packet (a different tag is inserted
depending on the used transmission ring). */

Gmac_Ip_SetTxOuterVlanTagForInsertion(INST_GMAC_0, 0U, GMAC_VLAN_TYPE_C_VLAN, 0x1U);

Gmac_Ip_SetTxOuterVlanTagForInsertion(INST_GMAC_0, 1U, GMAC_VLAN_TYPE_C_VLAN, 0x2U);

Gmac_Ip_SetTxOuterVlanTagForInsertion(INST_GMAC_0, 2U, GMAC_VLAN_TYPE_C_VLAN, 0x3U);

/* Set the inner VLAN tag to be replaced for each transmitted packet. */

Gmac_Ip_SetTxOuterVlanTagForReplacement(INST_GMAC_0, GMAC_VLAN_TYPE_S_VLAN, 0x03U);

/* Enable VLAN tag Rx filters. */

Gmac_Ip_SetVlanTagRxFilter(INST_GMAC_0, TRUE, &gmac_0_vlanRxFilterConfig);

/* Add different VLAN tags to the perfect filters. */

Gmac_Ip_AddVlanTagToRxFilter(INST_GMAC_0, 0U, 0x6003U);

Gmac_Ip_AddVlanTagToRxFilter(INST_GMAC_0, 1U, 0x6001U);

Gmac_Ip_AddVlanTagToRxFilter(INST_GMAC_0, 2U, 0x2U);

/* Add VLAN tags to the hash table for group filtering. */

Gmac_Ip_AddVlanTagToHashTable(INST_GMAC_0, 0x4U);

Gmac_Ip_AddVlanTagToHashTable(INST_GMAC_0, 0x5U);
```


Data Structures

- struct [Gmac_Ip_TxRingConfigType](#)
Specialization of ring configuration for Tx Rings. [More...](#)
- struct [Gmac_Ip_RxRingConfigType](#)
Specialization of ring configuration for Rx Rings. [More...](#)
- struct [Gmac_Ip_BufferType](#)
Send/Receive buffer information for the user. [More...](#)
- struct [Gmac_Ip_TimestampType](#)
Defines the timestamp. [More...](#)
- struct [Gmac_Ip_SysTimeConfigType](#)
Defines a structure to configure the system time. [More...](#)
- struct [Gmac_Ip_TxOptionsType](#)
Transmit options for a particular frame. [More...](#)
- struct [Gmac_Ip_TxInfoType](#)
Detailed status of a transmit buffer descriptor. [More...](#)
- struct [Gmac_Ip_RxInfoType](#)
Detailed status of a receive buffer descriptor. [More...](#)
- struct [Gmac_Ip_VlanConfigType](#)
Defines the VLAN configuration structure. [More...](#)

Macros

- `#define GMAC_TX_ERR_IP_HEADER_ERROR`
When set, it indicates that the Checksum Offload engine detected an IP header error.
- `#define GMAC_TX_ERR_JABBER_TIMEOUT`
When set, it indicates that the MAC transmitter has experienced a jabber time-out.
- `#define GMAC_TX_ERR_PACKET_FLUSHED`
When set, it indicates that the GMAC module flushed the packet because of a software flush command given by the CPU.
- `#define GMAC_TX_ERR_PAYLOAD_CHECKSUM_ERROR`
When set, it indicates that the Checksum Offload engine had a failure and did not insert any checksum into the encapsulated TCP, UDP, or ICMP payload.
- `#define GMAC_TX_ERR_LOSS_OF_CARRIER`
When set, it indicates that Loss of Carrier occurred during packet transmission. This is valid only for the packets transmitted without collision and when the MAC operates in the half-duplex mode.
- `#define GMAC_TX_ERR_NO_CARRIER`
When set, it indicates that the carrier sense signal from the PHY was not asserted during transmission.
- `#define GMAC_TX_ERR_LATE_COLLISION`
When set, it indicates that packet transmission was aborted because a collision occurred after the collision window (64 byte times including Preamble in MII mode and 512 byte times including Preamble and Carrier Extension in GMII mode). This bit is not valid if Underflow Error is set.
- `#define GMAC_TX_ERR_EXCESSIVE_COLLISION`
When set, it indicates that the transmission was aborted after 16 successive collisions while attempting to transmit the current packet.
- `#define GMAC_TX_ERR_EXCESSIVE_DEFERRAL`

When set, it indicates that the transmission ended because of excessive deferral of over 24,288 bit times (155,680 bits times in 1000 Mbps mode or Jumbo Packet enabled mode) if deferral check was set at init.

- #define [GMAC_TX_ERR_UNDERFLOW_ERROR](#)

When set, it indicates that the MAC aborted the packet because the data arrived late from the system memory.

- #define [GMAC_RX_ERR_CRC_ERROR](#)

When this bit is set, it indicates that a CRC Error occurred on the received packet.

- #define [GMAC_RX_ERR_DRIBBLE_ERROR](#)

When this bit is set, it indicates that the received packet has a non-integer multiple of bytes (odd nibbles). This bit is valid only in the MII Mode.

- #define [GMAC_RX_ERR_RECEIVE_ERROR](#)

When this bit is set, it indicates that the Receive error signal and Receive data valid signal are asserted during packet reception. This error also includes carrier extension error in the GMII and half-duplex mode.

- #define [GMAC_RX_ERR_WATCHDOG_TIMEOUT](#)

When this bit is set, it indicates that the Receive Watchdog Timer has expired while receiving the current packet. The current packet is truncated after watchdog timeout.

- #define [GMAC_RX_ERR_OVERFLOW_ERROR](#)

When this bit is set, it indicates that the received packet is damaged because of buffer overflow in Rx FIFO.

- #define [GMAC_RX_ERR_GIANT_PACKET](#)

When this bit is set, it indicates that the packet length exceeds the specified maximum Ethernet size of 1518, 1522, or 2000 bytes (9018 or 9022 bytes if jumbo packet enable is set).

Types Reference

- typedef void(* [Gmac_Ip_CallbackType](#)) (const uint8 Instance)
Callback function invoked when a general event is encountered.
- typedef void(* [Gmac_Ip_ChCallbackType](#)) (const uint8 Instance, const uint8 Channel)
Callback function invoked when a channel event is encountered.

Enum Reference

- enum [Gmac_Ip_StatusType](#)
Return values.
- enum [Gmac_Ip_CounterType](#)
Statistics counters enumeration.
- enum [Gmac_Ip_PowerStateType](#)
Controller power state indication.
- enum [Gmac_Ip_SpeedType](#)
Media Independent Interface speed selection.
- enum [Gmac_Ip_MacAddrFilterType](#)
Source/Destination address filter selection.
- enum [Gmac_Ip_CrcPadControlType](#)
CRC and Pad insertion control.
- enum [Gmac_Ip_ChecksumInsControlType](#)
Checksum insertion control.
- enum [Gmac_Ip_PayloadType](#)
Type of payload encapsulated in the IP datagram.

- enum [Gmac_Ip_VlanType](#)
VLAN type.
- enum [Gmac_Ip_VlanInsControlType](#)
VLAN tag and type deletion, insertion, or replacement control for transmitted packets.
- enum [Gmac_Ip_VlanStripControlType](#)
VLAN tag stripping control for received packets.
- enum [Gmac_Ip_SysTimeCorrOffsetType](#)
System time correction offset.
- enum [Gmac_Ip_TxSchedAlgoType](#)
Transmitter scheduling algorithm.
- enum [Gmac_Ip_TxThresholdType](#)
Available Tx threshold values.

Initialization and De-initialization

- [Gmac_Ip_StatusType Gmac_Ip_Init](#) (uint8 Instance, const [Gmac_CtrlConfigType](#) *Config)
Initializes the GMAC module.
- void [Gmac_Ip_Deinit](#) (uint8 Instance)
Deinitializes the GMAC module.
- [Gmac_Ip_PowerStateType Gmac_Ip_GetPowerState](#) (uint8 Instance)
Gets the current power state of the GMAC module.
- void [Gmac_Ip_SetPowerState](#) (uint8 Instance, [Gmac_Ip_PowerStateType](#) PowerState)
Sets the power state of the GMAC module.
- void [Gmac_Ip_EnableController](#) (uint8 Instance)
Enables all configured transmit and receive buffers and then enables the controller.
- [Gmac_Ip_StatusType Gmac_Ip_DisableController](#) (uint8 Instance)
Disables the controller and resets all the configured transmit and receive buffers.
- void [Gmac_Ip_SetSpeed](#) (uint8 Instance, [Gmac_Ip_SpeedType](#) Speed)
Sets the speed of the MII interface.
- [Gmac_Ip_StatusType Gmac_Ip_GetTxBuff](#) (uint8 Instance, uint8 Ring, [Gmac_Ip_BufferType](#) *Buff, uint16 *BuffId)
Provides a transmit buffer to be used by the application for transmission.
- [Gmac_Ip_StatusType Gmac_Ip_SendFrame](#) (uint8 Instance, uint8 Ring, const [Gmac_Ip_BufferType](#) *Buff, const [Gmac_Ip_TxOptionsType](#) *Options)
Sends an Ethernet frame.
- [Gmac_Ip_StatusType Gmac_Ip_SendMultiBufferFrame](#) (uint8 Instance, uint8 Ring, const [Gmac_Ip_BufferType](#) Buffers[], const [Gmac_Ip_TxOptionsType](#) *Options, uint32 NumBuffers)
Sends an Ethernet frame composed out of multiple buffers (not necessarily contiguous)
- [Gmac_Ip_StatusType Gmac_Ip_GetTxMultiBuff](#) (uint8 Instance, uint8 ring, uint16 NumBuffers, const uint16 BufferLength[], uint16 *buffId)
Provides a transmit buffer to be used by the application for transmission.
- [Gmac_Ip_StatusType Gmac_Ip_ReadFrame](#) (uint8 Instance, uint8 Ring, [Gmac_Ip_BufferType](#) *Buff, [Gmac_Ip_RxInfoType](#) *Info)
Reads a received Ethernet frame.
- void [Gmac_Ip_ProvideRxBuff](#) (uint8 Instance, uint8 Ring, const [Gmac_Ip_BufferType](#) *Buff)
Provides a receive buffer to be used by the driver for reception.

- boolean [Gmac_Ip_IsFrameAvailable](#) (uint8 Instance, uint8 Ring)
Checks if there are more frames available in the given queue.
- [Gmac_Ip_StatusType Gmac_Ip_GetTransmitStatus](#) (uint8 Instance, uint8 Ring, const [Gmac_Ip_BufferType](#) *Buff, [Gmac_Ip_TxInfoType](#) *Info)
Checks if the transmission of a buffer is complete.
- void [Gmac_Ip_SetRxExternalBuffer](#) (uint8 Instance, uint8 Ring, const [Gmac_Ip_BufferType](#) *Buff)
- uint32 [Gmac_Ip_GetCounter](#) (uint8 Instance, [Gmac_Ip_CounterType](#) Counter)
Gets statistics from the specified counter.
- void [Gmac_Ip_EnableMDIO](#) (uint8 Instance, boolean MiiPreambleDisabled, uint32 ModuleClk)
Enables the MDIO interface.
- [Gmac_Ip_StatusType Gmac_Ip_MDIORead](#) (uint8 Instance, uint8 PhyAddr, uint8 PhyReg, uint16 *Data, uint32 TimeoutMs)
Reads the selected register of the PHY.
- [Gmac_Ip_StatusType Gmac_Ip_MDIOWrite](#) (uint8 Instance, uint8 PhyAddr, uint8 PhyReg, uint16 Data, uint32 TimeoutMs)
Writes the selected register of the PHY.
- [Gmac_Ip_StatusType Gmac_Ip_MDIOReadMMD](#) (uint8 Instance, uint8 PhyAddr, uint8 Mmd, uint16 Phy↔Reg, uint16 *Data, uint32 TimeoutMs)
Reads a register of the specified MMD in a PHY device.
- [Gmac_Ip_StatusType Gmac_Ip_MDIOWriteMMD](#) (uint8 Instance, uint8 PhyAddr, uint8 Mmd, uint16 PhyReg, uint16 Data, uint32 TimeoutMs)
Writes a register of the specified MMD in a PHY device.
- void [Gmac_Ip_SetMacAddr](#) (uint8 Instance, const uint8 *MacAddr)
Configures the physical address of the MAC.
- void [Gmac_Ip_GetMacAddr](#) (uint8 Instance, uint8 *MacAddr)
Gets the physical address of the MAC.
- [Gmac_Ip_StatusType Gmac_Ip_TxTimeAwareShaperInit](#) (uint8 Instance, const [Gmac_CtrlConfigType](#) *Config)
Initialize time aware shaper.
- uint32 [Gmac_Ip_GetInterruptFlags](#) (uint8 Instance)
Gets a mask of the common interrupt events which occurred.
- uint32 [Gmac_Ip_GetChInterruptFlags](#) (uint8 Instance, uint8 Channel)
Gets a mask of the interrupt events associated to a channel which occurred.
- void [Gmac_Ip_SetBroadcastForwardAll](#) (uint8 Instance, boolean Enable)
Enables/Disables forwarding of the broadcast traffic.
- void [Gmac_Ip_SetMulticastForwardAll](#) (uint8 Instance, boolean Enable)
Enables/Disables forwarding of the multicast traffic, irrespective of the destination MAC address.
- void [Gmac_Ip_SetUnicastHashFilter](#) (uint8 Instance, boolean Enable)
Enables/Disables hash filtering for unicast traffic.
- void [Gmac_Ip_SetMulticastHashFilter](#) (uint8 Instance, boolean Enable)
Enables/Disables hash filtering for multicast traffic.
- void [Gmac_Ip_SetHashOrPerfectFilter](#) (uint8 Instance, boolean Enable)
Enables/Disables either hash or perfect filters. If the destination address of an incoming packet passes either the perfect filter or the hash filter, the packet is forwarded to the application.
- void [Gmac_Ip_AddDstAddrToHashFilter](#) (uint8 Instance, const uint8 *MacAddr)
Adds a hardware address to the hash filter. The destination address of an incoming packet is passed through CRC logic and then compared to the entries in the hash table.
- void [Gmac_Ip_RemoveDstAddrFromHashFilter](#) (uint8 Instance, const uint8 *MacAddr)

Removes a hardware address from the hash filter. The destination address of an incoming packet is passed through CRC logic and then compared to the entries in the hash table.

- void [Gmac_Ip_SetAddrPerfectFilter](#) (uint8 Instance, const uint8 *MacAddr, const uint8 Mask, const [Gmac_Ip_MacAddrFilterType](#) AddrType, boolean Enable)

Sets the second destination/source address perfect filter.

- void [Gmac_Ip_EnableVlan](#) (uint8 Instance, const [Gmac_Ip_VlanConfigType](#) *VlanConfig)

Enable and set VLAN control for transmitted and received packets.

- [Gmac_Ip_StatusType](#) [Gmac_Ip_SetTxOuterVlanTagForInsertion](#) (uint8 Instance, uint8 Ring, [Gmac_Ip_VlanType](#) VlanType, uint16 VlanTag)

Sets outer VLAN type and tag to be inserted by a particular transmission ring.

- [Gmac_Ip_StatusType](#) [Gmac_Ip_SetTxOuterVlanTagForReplacement](#) (uint8 Instance, [Gmac_Ip_VlanType](#) VlanType, uint16 VlanTag)

Sets outer VLAN tag to be replaced for all transmission rings.

- void [Gmac_Ip_SetTxInnerVlanTag](#) (uint8 Instance, [Gmac_Ip_VlanType](#) VlanType, uint16 VlanTag)

Sets inner VLAN type and tag to be inserted/replaced on transmission.

- void [Gmac_Ip_AddVlanTagToHashTable](#) (uint8 Instance, uint16 VlanTag)

Adds a VLAN Tag to the Hash Table filter.

- void [Gmac_Ip_RemoveVlanTagFromHashTable](#) (uint8 Instance, uint16 VlanTag)

Removes a VLAN Tag from the hash table filter.

- void [Gmac_Ip_InitSysTime](#) (uint8 Instance, const [Gmac_Ip_SysTimeConfigType](#) *SysTimeConfig)

Initialize system time.

- [Gmac_Ip_StatusType](#) [Gmac_Ip_SetSysTimeCorr](#) (uint8 Instance, [Gmac_Ip_SysTimeCorrOffsetType](#) Offset, uint32 SecondsUpdate, uint32 NanoSecondsUpdate)

Set system time correction.

- [Gmac_Ip_StatusType](#) [Gmac_Ip_SetRateRatioCorr](#) (uint8 Instance, float64 RateRatio)

Set system time correction.

- void [Gmac_Ip_GetSysTime](#) (uint8 Instance, [Gmac_Ip_TimestampType](#) *Timestamp)

Gets the current system time.

- void [Gmac_Ip_SetTxSchedAlgo](#) (uint8 Instance, [Gmac_Ip_TxSchedAlgoType](#) SchedAlgo)

Sets the transmitter scheduling algorithm.

- void [Gmac_Ip_SetTxRingWeight](#) (uint8 Instance, uint8 Ring, uint32 Weight)

Sets the weight (in WRR scheduling algorithm) for a particular Tx ring.

- void [Gmac_Ip_EnableTxStoreAndForward](#) (uint8 Instance, uint8 Ring)

Enables the store and forward feature on the transmit path.

- void [Gmac_Ip_SetTxThreshold](#) (uint8 Instance, uint8 Ring, [Gmac_Ip_TxThresholdType](#) ThresholdValue)

Sets transmit threshold levels.

- #define **ETH_43_GMAC_STOP_SEC_CODE**

7.2.2 Data Structure Documentation

7.2.2.1 struct [Gmac_Ip_TxRingConfigType](#)

Specialization of ring configuration for Tx Rings.

Definition at line 574 of file [Gmac_Ip_Types.h](#).

Data Fields

	Type	Name	Description
	uint32	Weight	Scheduling weight/quantum for DCB queues
	uint32	IdleSlopeCredit	Idle slope credit for AVB queues
	uint32	SendSlopeCredit	Send slope credit for AVB queues
	uint32	HiCredit	High credit limit for AVB queues
	sint32	LoCredit	Low credit limit for AVB queues
Gmac_Ip_BufferDescriptorType *		RingDesc	Buffer descriptor ring start address.
Gmac_Ip_ChCallbackType		Callback	Callback function for current channel.
uint8 *		Buffer	Buffer data pool start address
uint32		Interrupts	Channel interrupt sources. A logical OR of "Gmac_Ip_ChInterruptType".
uint16		BufferLen	Length of each individual buffer in a pool
uint16		RingSize	Buffer descriptors number.
uint8		PriorityMask	A logical OR of "Gmac_Ip_VlanPriorityType".
uint8		DmaBurstLength	Maximum number of beats in one DMA block data transfer.
Gmac_Ip_OperationModeType		QueueOpMode	Queue operation mode

7.2.2.2 struct Gmac_Ip_RxRingConfigType

Specialization of ring configuration for Rx Rings.

Definition at line 597 of file Gmac_Ip_Types.h.

Data Fields

- Gmac_Ip_BufferDescriptorType * [RingDesc](#)
- [Gmac_Ip_ChCallbackType](#) [Callback](#)
- uint8 * [Buffer](#)
- uint32 [Interrupts](#)
- uint16 [BufferLen](#)
- uint16 [RingSize](#)
- uint8 [PriorityMask](#)
- uint8 [DmaBurstLength](#)

7.2.2.2.1 Field Documentation

7.2.2.2.1.1 RingDesc Gmac_Ip_BufferDescriptorType* RingDesc

Buffer descriptor ring start address.

Definition at line 599 of file Gmac_Ip_Types.h.

7.2.2.2.1.2 Callback `Gmac_Ip_ChCallbackType` Callback

Callback function for current channel.

Definition at line 600 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.3 Buffer `uint8* Buffer`

Buffer data pool start address

Definition at line 601 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.4 Interrupts `uint32 Interrupts`

Channel interrupt sources. A logical OR of "Gmac_Ip_ChInterruptType".

Definition at line 602 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.5 BufferLen `uint16 BufferLen`

Length of each individual buffer in a pool

Definition at line 603 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.6 RingSize `uint16 RingSize`

Buffer descriptors number.

Definition at line 604 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.7 PriorityMask `uint8 PriorityMask`

A logical OR of "Gmac_Ip_VlanPriorityType".

Definition at line 605 of file `Gmac_Ip_Types.h`.

7.2.2.2.1.8 DmaBurstLength `uint8 DmaBurstLength`

Maximum number of beats in one DMA block data transfer.

Definition at line 606 of file Gmac_Ip_Types.h.

7.2.2.3 struct Gmac_Ip_BufferType

Send/Receive buffer information for the user.

Definition at line 614 of file Gmac_Ip_Types.h.

Data Fields

- `uint8 * Data`
- `uint16 Length`

7.2.2.3.1 Field Documentation

7.2.2.3.1.1 Data `uint8* Data`

Pointer to the data buffer

Definition at line 616 of file Gmac_Ip_Types.h.

7.2.2.3.1.2 Length `uint16 Length`

Length of the data buffer

Definition at line 617 of file Gmac_Ip_Types.h.

7.2.2.4 struct Gmac_Ip_TimestampType

Defines the timestamp.

Definition at line 624 of file Gmac_Ip_Types.h.

Data Fields

- `uint32 nanoseconds`
- `uint32 seconds`
- `uint16 secondsHi`

7.2.2.4.1 Field Documentation

7.2.2.4.1.1 nanoseconds `uint32 nanoseconds`

Nanoseconds part of the time.

Definition at line 626 of file `Gmac_Ip_Types.h`.

7.2.2.4.1.2 seconds `uint32 seconds`

The 32 least significant bits of the 48 bits seconds part of the time.

Definition at line 627 of file `Gmac_Ip_Types.h`.

7.2.2.4.1.3 secondsHi `uint16 secondsHi`

The 16 most significant bits of the 48 bit seconds part of the time.

Definition at line 628 of file `Gmac_Ip_Types.h`.

7.2.2.5 struct `Gmac_Ip_SysTimeConfigType`

Defines a structure to configure the system time.

Definition at line 635 of file `Gmac_Ip_Types.h`.

Data Fields

- `uint8 SubSecondInc`
- `uint8 SubNanoSecondsInc`
- `Gmac_Ip_TimestampType * InitialTimestamp`

7.2.2.5.1 Field Documentation

7.2.2.5.1.1 SubSecondInc `uint8 SubSecondInc`

The value in nanoseconds to be accumulated every clock cycle.

Definition at line 637 of file `Gmac_Ip_Types.h`.

7.2.2.5.1.2 SubNanoSecondsInc `uint8 SubNanoSecondsInc`

The value in sub-nanoseconds to be accumulated every clock cycle, represented in nanoseconds multiplied by 2^8 . For example, if the required increment is 5.3ns, then subSecondInc should be 0x05 and subNanosecondsInc should be 0x4C ($0.3 * 2^8$).

Definition at line 638 of file Gmac_Ip_Types.h.

7.2.2.5.1.3 InitialTimestamp `Gmac_Ip_TimestampType* InitialTimestamp`

Nanoseconds part of the time.

Definition at line 641 of file Gmac_Ip_Types.h.

7.2.2.6 struct Gmac_Ip_TxOptionsType

Transmit options for a particular frame.

Definition at line 648 of file Gmac_Ip_Types.h.

Data Fields

- boolean `NoInt`
- `Gmac_Ip_CrcPadControlType CrcPadIns`
- `Gmac_Ip_ChecksumInsControlType ChecksumIns`

7.2.2.6.1 Field Documentation

7.2.2.6.1.1 NoInt `boolean NoInt`

Does not generate interrupt on transmit completion.

Definition at line 650 of file Gmac_Ip_Types.h.

7.2.2.6.1.2 CrcPadIns `Gmac_Ip_CrcPadControlType CrcPadIns`

Controls the CRC and pad insertion for Tx packet.

Definition at line 651 of file Gmac_Ip_Types.h.

7.2.2.6.1.3 ChecksumIns `Gmac_Ip_ChecksumInsControlType` ChecksumIns

Controls the IP checksum insertion for Tx packet.

Definition at line 652 of file `Gmac_Ip_Types.h`.

7.2.2.7 struct `Gmac_Ip_TxInfoType`

Detailed status of a transmit buffer descriptor.

Definition at line 659 of file `Gmac_Ip_Types.h`.

Data Fields

- uint32 `ErrMask`
- `Gmac_Ip_TimestampType` Timestamp

7.2.2.7.1 Field Documentation

7.2.2.7.1.1 ErrMask `uint32` ErrMask

Mask of `GMAC_TX_ERR_*`, representing the errors that occurred.

Definition at line 661 of file `Gmac_Ip_Types.h`.

7.2.2.7.1.2 Timestamp `Gmac_Ip_TimestampType` Timestamp

The timestamp of the transmitted packet.

Definition at line 662 of file `Gmac_Ip_Types.h`.

7.2.2.8 struct `Gmac_Ip_RxInfoType`

Detailed status of a receive buffer descriptor.

Definition at line 711 of file `Gmac_Ip_Types.h`.

Data Fields

- uint32 [ErrMask](#)
- uint16 [PktLen](#)
- [Gmac_Ip_PayloadType](#) PayloadType
- boolean [Ipv6](#)
- boolean [Ipv4](#)
- boolean [IpPayloadErr](#)
- boolean [IpHeaderErr](#)
- uint16 [OuterVlanTag](#)
- uint16 [InnerVlanTag](#)
- [Gmac_Ip_TimestampType](#) Timestamp

7.2.2.8.1 Field Documentation

7.2.2.8.1.1 ErrMask `uint32 ErrMask`

Mask of `GMAC_RX_ERR_*`, representing the errors that occurred.

Definition at line 713 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.2 PktLen `uint16 PktLen`

Byte length of the received packet that was transferred to system memory (including CRC).

Definition at line 714 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.3 PayloadType `Gmac_Ip_PayloadType PayloadType`

The type of payload encapsulated in the IP datagram (this field is valid only if the Checksum Offloading feature is enabled when initializing the GMAC module).

Definition at line 715 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.4 Ipv6 `boolean Ipv6`

Indicates that the frame is an IPv6 frame (this field is valid only if the Checksum Offloading feature is enabled when initializing the GMAC module).

Definition at line 717 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.5 Ipv4 `boolean Ipv4`

Indicates that the frame is an IPv4 frame (this field is valid only if the Checksum Offloading feature is enabled when initializing the GMAC module).

Definition at line 719 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.6 IpPayloadErr `boolean IpPayloadErr`

Indicates that an IP payload error has occurred (this field is valid only if the Checksum Offloading feature is enabled when initializing the GMAC module).

Definition at line 721 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.7 IpHeaderErr `boolean IpHeaderErr`

Indicates that an IP header error has occurred (this field is valid only if the Checksum Offloading feature is enabled when initializing the GMAC module).

Definition at line 723 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.8 OuterVlanTag `uint16 OuterVlanTag`

Contains the outer VLAN tag of the received packet.

Definition at line 725 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.9 InnerVlanTag `uint16 InnerVlanTag`

Contains the inner VLAN tag of the received packet. This is valid only when Double VLAN tag processing is enabled.

Definition at line 726 of file `Gmac_Ip_Types.h`.

7.2.2.8.1.10 Timestamp `Gmac_Ip_TimestampType Timestamp`

The timestamp of the received packet.

Definition at line 728 of file `Gmac_Ip_Types.h`.

7.2.2.9 struct Gmac_Ip_VlanConfigType

Defines the VLAN configuration structure.

Definition at line 759 of file Gmac_Ip_Types.h.

Data Fields

- boolean [EnDoubleVlan](#)
- boolean [EnSvlan](#)
- [Gmac_Ip_VlanInsControlType](#) OuterVlanIns
- [Gmac_Ip_VlanInsControlType](#) InnerVlanIns
- [Gmac_Ip_VlanStripControlType](#) OuterVlanStrip
- [Gmac_Ip_VlanStripControlType](#) InnerVlanStrip

7.2.2.9.1 Field Documentation

7.2.2.9.1.1 EnDoubleVlan `boolean EnDoubleVlan`

Enables processing of up to two VLAN tags on Tx and Rx (if present).

Definition at line 761 of file Gmac_Ip_Types.h.

7.2.2.9.1.2 EnSvlan `boolean EnSvlan`

Enables MAC transmitter and receiver to consider the S-VLAN packets (Type = 0x88A8) as valid VLAN tagged packets.

Definition at line 762 of file Gmac_Ip_Types.h.

7.2.2.9.1.3 OuterVlanIns `Gmac_Ip_VlanInsControlType OuterVlanIns`

Selects if outer VLAN tag and type are deleted, inserted or replaced in transmitted packets.

Definition at line 763 of file Gmac_Ip_Types.h.

7.2.2.9.1.4 InnerVlanIns `Gmac_Ip_VlanInsControlType InnerVlanIns`

Selects if inner VLAN tag and type are deleted, inserted or replaced in transmitted packets. Effective only if double VLAN processing is enabled.

Definition at line 764 of file Gmac_Ip_Types.h.

7.2.2.9.1.5 OuterVlanStrip `Gmac_Ip_VlanStripControlType` OuterVlanStrip

Selects if outer VLAN tag is stripped in received packets.

Definition at line 766 of file Gmac_Ip_Types.h.

7.2.2.9.1.6 InnerVlanStrip `Gmac_Ip_VlanStripControlType` InnerVlanStrip

Selects if inner VLAN tag is stripped in received packets.

Definition at line 767 of file Gmac_Ip_Types.h.

7.2.3 Macro Definition Documentation**7.2.3.1 GMAC_TX_ERR_IP_HEADER_ERROR**

```
#define GMAC_TX_ERR_IP_HEADER_ERROR
```

When set, it indicates that the Checksum Offload engine detected an IP header error.

Definition at line 666 of file Gmac_Ip_Types.h.

7.2.3.2 GMAC_TX_ERR_JABBER_TIMEOUT

```
#define GMAC_TX_ERR_JABBER_TIMEOUT
```

When set, it indicates that the MAC transmitter has experienced a jabber time-out.

Definition at line 669 of file Gmac_Ip_Types.h.

7.2.3.3 GMAC_TX_ERR_PACKET_FLUSHED

```
#define GMAC_TX_ERR_PACKET_FLUSHED
```

When set, it indicates that the GMAC module flushed the packet because of a software flush command given by the CPU.

Definition at line 673 of file Gmac_Ip_Types.h.

7.2.3.4 GMAC_TX_ERR_PAYLOAD_CHECKSUM_ERROR

```
#define GMAC_TX_ERR_PAYLOAD_CHECKSUM_ERROR
```

When set, it indicates that the Checksum Offload engine had a failure and did not insert any checksum into the encapsulated TCP, UDP, or ICMP payload.

Definition at line 677 of file Gmac_Ip_Types.h.

7.2.3.5 GMAC_TX_ERR_LOSS_OF_CARRIER

```
#define GMAC_TX_ERR_LOSS_OF_CARRIER
```

When set, it indicates that Loss of Carrier occurred during packet transmission. This is valid only for the packets transmitted without collision and when the MAC operates in the half-duplex mode.

Definition at line 682 of file Gmac_Ip_Types.h.

7.2.3.6 GMAC_TX_ERR_NO_CARRIER

```
#define GMAC_TX_ERR_NO_CARRIER
```

When set, it indicates that the carrier sense signal from the PHY was not asserted during transmission.

Definition at line 686 of file Gmac_Ip_Types.h.

7.2.3.7 GMAC_TX_ERR_LATE_COLLISION

```
#define GMAC_TX_ERR_LATE_COLLISION
```

When set, it indicates that packet transmission was aborted because a collision occurred after the collision window (64 byte times including Preamble in MII mode and 512 byte times including Preamble and Carrier Extension in GMII mode). This bit is not valid if Underflow Error is set.

Definition at line 692 of file Gmac_Ip_Types.h.

7.2.3.8 GMAC_TX_ERR_EXCESSIVE_COLLISION

```
#define GMAC_TX_ERR_EXCESSIVE_COLLISION
```

When set, it indicates that the transmission was aborted after 16 successive collisions while attempting to transmit the current packet.

Definition at line 696 of file Gmac_Ip_Types.h.

7.2.3.9 GMAC_TX_ERR_EXCESSIVE_DEFERRAL

```
#define GMAC_TX_ERR_EXCESSIVE_DEFERRAL
```

When set, it indicates that the transmission ended because of excessive deferral of over 24,288 bit times (155,680 bits times in 1000 Mbps mode or Jumbo Packet enabled mode) if deferral check was set at init.

Definition at line 701 of file Gmac_Ip_Types.h.

7.2.3.10 GMAC_TX_ERR_UNDERFLOW_ERROR

```
#define GMAC_TX_ERR_UNDERFLOW_ERROR
```

When set, it indicates that the MAC aborted the packet because the data arrived late from the system memory.

Definition at line 705 of file Gmac_Ip_Types.h.

7.2.3.11 GMAC_RX_ERR_CRC_ERROR

```
#define GMAC_RX_ERR_CRC_ERROR
```

When this bit is set, it indicates that a CRC Error occurred on the received packet.

Definition at line 732 of file Gmac_Ip_Types.h.

7.2.3.12 GMAC_RX_ERR_DRIBBLE_ERROR

```
#define GMAC_RX_ERR_DRIBBLE_ERROR
```

When this bit is set, it indicates that the received packet has a non-integer multiple of bytes (odd nibbles). This bit is valid only in the MII Mode.

Definition at line 736 of file Gmac_Ip_Types.h.

7.2.3.13 GMAC_RX_ERR_RECEIVE_ERROR

```
#define GMAC_RX_ERR_RECEIVE_ERROR
```

When this bit is set, it indicates that the Receive error signal and Receive data valid signal are asserted during packet reception. This error also includes carrier extension error in the GMII and half-duplex mode.

Definition at line 741 of file Gmac_Ip_Types.h.

7.2.3.14 GMAC_RX_ERR_WATCHDOG_TIMEOUT

```
#define GMAC_RX_ERR_WATCHDOG_TIMEOUT
```

When this bit is set, it indicates that the Receive Watchdog Timer has expired while receiving the current packet. The current packet is truncated after watchdog timeout.

Definition at line 745 of file Gmac_Ip_Types.h.

7.2.3.15 GMAC_RX_ERR_OVERFLOW_ERROR

```
#define GMAC_RX_ERR_OVERFLOW_ERROR
```

When this bit is set, it indicates that the received packet is damaged because of buffer overflow in Rx FIFO.

Definition at line 749 of file Gmac_Ip_Types.h.

7.2.3.16 GMAC_RX_ERR_GIANT_PACKET

```
#define GMAC_RX_ERR_GIANT_PACKET
```

When this bit is set, it indicates that the packet length exceeds the specified maximum Ethernet size of 1518, 1522, or 2000 bytes (9018 or 9022 bytes if jumbo packet enable is set).

Definition at line 753 of file Gmac_Ip_Types.h.

7.2.4 Types Reference

7.2.4.1 Gmac_Ip_CallbackType

```
typedef void(* Gmac_Ip_CallbackType) (const uint8 Instance)
```

Callback function invoked when a general event is encountered.

Definition at line 514 of file Gmac_Ip_Types.h.

7.2.4.2 Gmac_Ip_ChCallbackType

```
typedef void(* Gmac_Ip_ChCallbackType) (const uint8 Instance, const uint8 Channel)
```

Callback function invoked when a channel event is encountered.

Definition at line 519 of file Gmac_Ip_Types.h.

7.2.5 Enum Reference

7.2.5.1 Gmac_Ip_StatusType

```
enum Gmac_Ip_StatusType
```

Return values.

Enumerator

GMAC_STATUS_SUCCESS	Generic operation success status
GMAC_STATUS_ERROR	Generic operation failure status
GMAC_STATUS_BUSY	Generic operation busy status
GMAC_STATUS_TIMEOUT	Generic operation timeout status
GMAC_STATUS_UNSUPPORTED	Generic operation unsupported status
GMAC_STATUS_CACHE_ERROR	Generic operation cache operation failed
GMAC_STATUS_RX_QUEUE_EMPTY	There is no available frame in the receive queue
GMAC_STATUS_TX_QUEUE_FULL	There is no available space for the frame in the transmit queue
GMAC_STATUS_BUFF_NOT_FOUND	The specified buffer was not found in the queue
GMAC_STATUS_TX_BUFF_BUSY	All internal TX buffers are currently in use
GMAC_STATUS_TX_BUFF_OVERFLOW	The requested TX buffer length cannot

Definition at line 100 of file Gmac_Ip_Types.h.

7.2.5.2 Gmac_Ip_CounterType

enum `Gmac_Ip_CounterType`

Statistics counters enumeration.

Definition at line 236 of file `Gmac_Ip_Types.h`.

7.2.5.3 Gmac_Ip_PowerStateType

enum `Gmac_Ip_PowerStateType`

Controller power state indication.

Definition at line 297 of file `Gmac_Ip_Types.h`.

7.2.5.4 Gmac_Ip_SpeedType

enum `Gmac_Ip_SpeedType`

Media Independent Interface speed selection.

Enumerator

GMAC_SPEED_1G	Speed 1 Gbps.
GMAC_SPEED_10M	Speed 10 Mbps.
GMAC_SPEED_100M	Speed 100 Mbps.

Definition at line 309 of file `Gmac_Ip_Types.h`.

7.2.5.5 Gmac_Ip_MacAddrFilterType

enum `Gmac_Ip_MacAddrFilterType`

Source/Destination address filter selection.

Enumerator

GMAC_DST_ADDR_FILTER	Destination address filter.
GMAC_SRC_ADDR_FILTER	Source address filter.
GMAC_SRC_ADDR_INV_FILTER	Source address inverse filter.

S32K3XX ETH Driver

Definition at line 356 of file Gmac_Ip_Types.h.

7.2.5.6 Gmac_Ip_CrcPadControlType

enum `Gmac_Ip_CrcPadControlType`

CRC and Pad insertion control.

Enumerator

GMAC_CRC_AND_PAD_INSERTION	Appends the CRC at the end of the transmitted packet of length greater than or equal to 60 bytes. Appends padding and CRC to a packet with length less than 60 bytes.
GMAC_CRC_INSERTION	Appends the CRC at the end of the transmitted packet but it does not append padding. The application should ensure that the padding bytes are present in the packet being transferred.
GMAC_CRC_INSERTION_DISABLE	Does not append the CRC at the end of the transmitted packet. The application should ensure that the padding and CRC bytes are present in the packet being transferred.
GMAC_CRC_REPLACEMENT	Replaces the last four bytes of the transmitted packet with recalculated CRC bytes. The application should ensure that the padding and CRC bytes are present in the packet being transferred.

Definition at line 367 of file Gmac_Ip_Types.h.

7.2.5.7 Gmac_Ip_ChecksumInsControlType

enum `Gmac_Ip_ChecksumInsControlType`

Checksum insertion control.

Enumerator

GMAC_CHECKSUM_INSERTION_DISABLE	Checksum insertion is disabled.
GMAC_CHECKSUM_INSERTION_IP	IP header checksum calculation and insertion are enabled.
GMAC_CHECKSUM_INSERTION_PROTO_N← O_PSEUDOH	Protocol checksum calculation and insertion are enabled, but pseudo-header checksum is not calculated in hardware.
GMAC_CHECKSUM_INSERTION_PROTO_PS← EUDOH	Protocol checksum calculation and insertion are enabled, and pseudo-header checksum is calculated in hardware.

Definition at line 383 of file Gmac_Ip_Types.h.

7.2.5.8 Gmac_Ip_PayloadType

enum `Gmac_Ip_PayloadType`

Type of payload encapsulated in the IP datagram.

Enumerator

GMAC_IP_PAYLOAD_TYPE_UNKNOWN	Unknown type or IP payload not processed.
GMAC_IP_PAYLOAD_TYPE_UDP	UDP packet.
GMAC_IP_PAYLOAD_TYPE_TCP	TCP packet.
GMAC_IP_PAYLOAD_TYPE_ICMP	ICMP packet.

Definition at line 397 of file Gmac_Ip_Types.h.

7.2.5.9 Gmac_Ip_VlanType

enum `Gmac_Ip_VlanType`

VLAN type.

Enumerator

GMAC_VLAN_TYPE_C_VLAN	C-VLAN type (0x8100).
GMAC_VLAN_TYPE_S_VLAN	S-VLAN type (0x88A8).

Definition at line 409 of file Gmac_Ip_Types.h.

7.2.5.10 Gmac_Ip_VlanInsControlType

enum `Gmac_Ip_VlanInsControlType`

VLAN tag and type deletion, insertion, or replacement control for transmitted packets.

Enumerator

GMAC_VLAN_TAG_NO_CONTROL	The VLAN tag is not altered in the packet to be transmitted.
--------------------------	--

Enumerator

GMAC_VLAN_TAG_DELETION	The VLAN tag is deleted from the packet to be transmitted.
GMAC_VLAN_TAG_INSERTION	The VLAN tag is inserted in the packet to be transmitted.
GMAC_VLAN_TAG_REPLACEMENT	The VLAN tag is replaced in the packet to be transmitted.

Definition at line 419 of file Gmac_Ip_Types.h.

7.2.5.11 Gmac_Ip_VlanStripControlType

```
enum Gmac_Ip_VlanStripControlType
```

VLAN tag stripping control for received packets.

Enumerator

GMAC_VLAN_TAG_DO_NOT_STRIP	Does not strip the inner VLAN tag in the received packet.
GMAC_VLAN_TAG_STRIP_IF_FILTER_PASS	Strips the inner VLAN tag in the received packet if the VLAN filter passes.
GMAC_VLAN_TAG_STRIP_IF_FILTER_FAIL	Strips the inner VLAN tag in the received packet if the VLAN filter fails.
GMAC_VLAN_TAG_ALWAYS_STRIP	Always strips the inner VLAN tag in the received packet.

Definition at line 431 of file Gmac_Ip_Types.h.

7.2.5.12 Gmac_Ip_SysTimeCorrOffsetType

```
enum Gmac_Ip_SysTimeCorrOffsetType
```

System time correction offset.

Enumerator

GMAC_SYS_TIME_CORR_POSITIVE	Correction value is added to the system time.
GMAC_SYS_TIME_CORR_NEGATIVE	Correction value is subtracted from the system time.

Definition at line 460 of file Gmac_Ip_Types.h.

7.2.5.13 Gmac_Ip_TxSchedAlgoType

enum `Gmac_Ip_TxSchedAlgoType`

Transmitter scheduling algorithm.

Enumerator

GMAC_SCHED_ALGO_WRR	Weighted round robin algorithm.
GMAC_SCHED_ALGO_WFQ	Weighted fair queueing algorithm.
GMAC_SCHED_ALGO_DWRR	Deficit weighted round robin algorithm.
GMAC_SCHED_ALGO_SP	Strict priority algorithm.

Definition at line 470 of file `Gmac_Ip_Types.h`.

7.2.5.14 Gmac_Ip_TxThresholdType

enum `Gmac_Ip_TxThresholdType`

Available Tx threshold values.

Definition at line 481 of file `Gmac_Ip_Types.h`.

7.2.6 Function Reference

7.2.6.1 Gmac_Ip_Init()

```
Gmac_Ip_StatusType Gmac_Ip_Init (
    uint8 Instance,
    const Gmac_CtrlConfigType * Config )
```

Initializes the GMAC module.

This function initializes and enables the GMAC module, configuring receive and transmit control settings, the receive and transmit descriptors rings, and the MAC physical address.

Note: All untagged packets will be received by ring 0 (zero). VLAN tagged packets are routed depending on the VLAN Tag Priority field according to the provided configuration.

Parameters

in	<i>instance</i>	Instance number
in	<i>config</i>	Pointer to the module configuration structure

Return values

<i>GMAC_STATUS_SUCCESS</i>	The initialization was successful.
<i>GMAC_STATUS_TIMEOUT</i>	The DMA subsystem reset could not complete.

7.2.6.2 Gmac_Ip_Deinit()

```
void Gmac_Ip_Deinit (
    uint8 Instance )
```

Deinitializes the GMAC module.

This function disables the interrupts and then disables the GMAC module.

Parameters

in	<i>instance</i>	Instance number
----	-----------------	-----------------

7.2.6.3 Gmac_Ip_GetPowerState()

```
Gmac_Ip_PowerStateType Gmac_Ip_GetPowerState (
    uint8 Instance )
```

Gets the current power state of the GMAC module.

Parameters

in	<i>instance</i>	Instance number
----	-----------------	-----------------

Returns

Gmac_Ip_PowerStateType The power state of the controller

7.2.6.4 Gmac_Ip_SetPowerState()

```
void Gmac_Ip_SetPowerState (
    uint8 Instance,
    Gmac_Ip_PowerStateType PowerState )
```

Sets the power state of the GMAC module.

Parameters

in	<i>instance</i>	Instance number
in	<i>powerState</i>	Power state to transition into

7.2.6.5 Gmac_Ip_EnableController()

```
void Gmac_Ip_EnableController (  
    uint8 Instance )
```

Enables all configured transmit and receive buffers and then enables the controller.

Parameters

in	<i>instance</i>	Instance number
----	-----------------	-----------------

7.2.6.6 Gmac_Ip_DisableController()

```
Gmac_Ip_StatusType Gmac_Ip_DisableController (  
    uint8 Instance )
```

Disables the controller and resets all the configured transmit and receive buffers.

Warning: This function ignores all pending transmission and reception requests

Parameters

in	<i>instance</i>	Instance number
----	-----------------	-----------------

Return values

<i>GMAC_STATUS_SUCCESS</i>	The controller was successfully disabled.
<i>GMAC_STATUS_TIMEOUT</i>	The underlying MTL queues could not be flushed.

7.2.6.7 Gmac_Ip_SetSpeed()

```
void Gmac_Ip_SetSpeed (  
    uint8 Instance,  
    Gmac_Ip_SpeedType Speed )
```

Sets the speed of the MII interface.

Parameters

in	<i>instance</i>	Instance number
in	<i>speed</i>	MI speed

7.2.6.8 Gmac_Ip_GetTxBuff()

```
Gmac_Ip_StatusType Gmac_Ip_GetTxBuff (
    uint8 Instance,
    uint8 Ring,
    Gmac_Ip_BufferType * Buff,
    uint16 * BuffId )
```

Provides a transmit buffer to be used by the application for transmission.

This function provides an internal buffer which can further be used by the application to store the transmit data.

Note: The buffer will be marked as locked and won't be released until after a call to Gmac_Ip_GetTransmitStatus for the same buffer returns GMAC_STATUS_SUCCESS.

Important: The driver does not ensure synchronization between different threads trying to get a buffer at the same time. This synchronization shall be implemented by the application.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	Ring number
in, out	<i>buff</i>	In: Buffer containing the desired length Out: Buffer containing the granted length or available length in case of overflow.
out	<i>buffId</i>	Index of the buffer (descriptor) within the ring. If this information is not needed, this parameter should be NULL_PTR.

Return values

<i>GMAC_STATUS_SUCCESS</i>	The buffer has been successfully locked.
<i>GMAC_STATUS_TX_BUFF_BUSY</i>	All buffers are currently in use.
<i>GMAC_STATUS_TX_BUFF_OVERFLOW</i>	The requested buffer length cannot be granted.

7.2.6.9 Gmac_Ip_SendFrame()

```
Gmac_Ip_StatusType Gmac_Ip_SendFrame (
```

Module Documentation

```
uint8 Instance,
uint8 Ring,
const Gmac_Ip_BufferType * Buff,
const Gmac_Ip_TxOptionsType * Options )
```

Sends an Ethernet frame.

This function sends an Ethernet frame, contained in the buffer received as parameter.

Note: Since the transmission of the frame is not complete when this function returns, the application must not change/alter/re-use the provided buffer until after a call to Gmac_Ip_GetTransmitStatus for the same buffer returns GMAC_STATUS_SUCCESS.

Important: The driver does not ensure synchronization between different threads trying to send a frame at the same time. This synchronization shall be implemented by the application.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	The ring number
in	<i>buff</i>	The buffer containing the frame
in	<i>options</i>	Configuration options applicable to this buffer's transmission only. Can be NULL_PTR, if no special option is required.

Return values

<i>GMAC_STATUS_SUCCESS</i>	The frame was successfully enqueued for transmission.
<i>GMAC_STATUS_TX_QUEUE_FULL</i>	There is no available space for the frame in the queue.

7.2.6.10 Gmac_Ip_SendMultiBufferFrame()

```
Gmac_Ip_StatusType Gmac_Ip_SendMultiBufferFrame (
    uint8 Instance,
    uint8 Ring,
    const Gmac_Ip_BufferType Buffers[],
    const Gmac_Ip_TxOptionsType * Options,
    uint32 NumBuffers )
```

Sends an Ethernet frame composed out of multiple buffers (not necessarily contiguous)

This function sends an Ethernet frame contained in the buffers received as parameter.

Note: Since the transmission of the frame is not complete when this function returns, the application must not change/alter/re-use the provided buffers until after a call to Gmac_Ip_GetTransmitStatus for the same buffer returns GMAC_STATUS_SUCCESS.

Important: The driver does not ensure synchronization between different threads trying to send a frame at the same time. This synchronization shall be implemented by the application.

Parameters

in	<i>Instance</i>	Instance number
in	<i>Ring</i>	The ring number
in	<i>Buffers</i>	The contiguous buffer array composing the frame
in	<i>Options</i>	Configuration options applied to the transmitted frame. Must be different from NULL_PTR.
in	<i>NumBuffers</i>	The size of the Buffers array

Return values

<i>GMAC_STATUS_SUCCESS</i>	The frame was successfully enqueued for transmission.
<i>GMAC_STATUS_TX_QUEUE_FULL</i>	There is no available space for the frame in the queue.

7.2.6.11 Gmac_Ip_GetTxMultiBuff()

```
Gmac_Ip_StatusType Gmac_Ip_GetTxMultiBuff (
    uint8 Instance,
    uint8 ring,
    uint16 NumBuffers,
    const uint16 BufferLength[],
    uint16 * buffId )
```

Provides a transmit buffer to be used by the application for transmission.

This function will verify if there are enough descriptors free and that each of the descriptors can hold the parts of the frame to be send using Gmac_Ip_SendMultiBufferFrame.

Note: This function will only return the first buffer descriptor index starting with which the multi buffer frame can be sent.

Important: This function is meant for internal use only and will be called from an upper layer to get the first buffer descriptor index from a sequence of buffers that will be used for sending a multi buffer frame.

Parameters

in	<i>Instance</i>	Instance number
in	<i>ring</i>	Ring number
in	<i>NumBuffers</i>	Number of buffers
in	<i>BufferLength</i>	List with the length of each chunk of the frame
out	<i>buffId</i>	Index of the buffer (descriptor) within the ring.

Return values

<i>GMAC_STATUS_SUCCESS</i>	The buffer has been successfully locked.
----------------------------	--

Return values

<i>GMAC_STATUS_TX_BUFF_BUSY</i>	All buffers are currently in use for the current ring.
<i>GMAC_STATUS_TX_BUFF_OVERFLOW</i>	The requested buffer length cannot be granted.
<i>GMAC_STATUS_INVALID_FRAME_LENGTH</i>	The buffer length is smaller than minium of frame length.

7.2.6.12 Gmac_Ip_ReadFrame()

```
Gmac_Ip_StatusType Gmac_Ip_ReadFrame (
    uint8 Instance,
    uint8 Ring,
    Gmac_Ip_BufferType * Buff,
    Gmac_Ip_RxInfoType * Info )
```

Reads a received Ethernet frame.

This function reads the first received Ethernet frame in the Rx queue. The buffer received as parameter will be updated by the driver and the .data field will point to a memory area containing the frame data.

Note: Once the application finished processing the buffer, it could be reused by the driver for further receptions by invoking Gmac_Ip_ProvideRxBuff.

Important: The driver does not ensure synchronization between different threads trying to read a frame at the same time. This synchronization shall be implemented by the application.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	The ring number
out	<i>buff</i>	The buffer containing the frame
out	<i>info</i>	Enhanced information related to the data contained by this receive buffer. If this information is not needed, this parameter should be NULL_PTR.

Return values

<i>GMAC_STATUS_SUCCESS</i>	A frame was successfully read.
<i>GMAC_STATUS_RX_QUEUE_EMPTY</i>	There is no available frame in the queue.

7.2.6.13 Gmac_Ip_ProvideRxBuff()

```
void Gmac_Ip_ProvideRxBuff (
    uint8 Instance,
```

```
uint8 Ring,
const Gmac_Ip_BufferType * Buff )
```

Provides a receive buffer to be used by the driver for reception.

This function provides a buffer which can further be used by the reception mechanism in order to store the received data.

Note: The application can either provide a buffer previously obtained in a Gmac_Ip_ReadFrame call (when it is no longer needed after being fully processed), or allocate a new buffer. The former approach is recommended as it has a simpler usage model and re-uses the same initial memory range for the entire driver lifetime operation. The later approach could provide more flexibility, but since it involves constant memory free/alloc operations it is only recommended with an efficient pool-based memory allocator.

Important: The driver does not ensure synchronization between different threads trying to provide a buffer at the same time. This synchronization shall be implemented by the application.

Important: The application is responsible for providing one Rx buffer for every frame it receives, otherwise the reception ring can fill-up, affecting further reception.

Usage example:

```
stat = Gmac_Ip_ReadFrame(INST_GMAC1, 0U, &rxBuff);

if (stat == GMAC_STATUS_SUCCESS) { process_buffer(&rxBuff); Gmac_Ip_ProvideRxBuff(INST_GMAC1,
0U, &rxBuff); }
```

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	The ring number
in	<i>buff</i>	The buffer to be added to the reception ring

7.2.6.14 Gmac_Ip_IsFrameAvailable()

```
boolean Gmac_Ip_IsFrameAvailable (
    uint8 Instance,
    uint8 Ring )
```

Checks if there are more frames available in the given queue.

This function takes a peek at the given Rx queue to check if there are more Ethernet frames to be received. Its intended usage is to provide this information without also extracting the frame as "Gmac_Ip_ReadFrame".

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	The ring number

Return values

<i>TRUE</i>	There is an available frame in the queue.
<i>FALSE</i>	There is no available frame in the queue.

7.2.6.15 Gmac_Ip_GetTransmitStatus()

```
Gmac_Ip_StatusType Gmac_Ip_GetTransmitStatus (
    uint8 Instance,
    uint8 Ring,
    const Gmac_Ip_BufferType * Buff,
    Gmac_Ip_TxInfoType * Info )
```

Checks if the transmission of a buffer is complete.

This function checks if the transmission of the given buffer is complete.

Note: If the buffer is not found in the Tx ring, the function will return GMAC_STATUS_BUFF_NOT_FOUND.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	The ring number
in	<i>buff</i>	The transmit buffer for which the status shall be checked
out	<i>info</i>	Extended information related to the buffer. If this information is not needed, this parameter should be NULL_PTR.

Return values

<i>GMAC_STATUS_BUSY</i>	The frame is still enqueued for transmission.
<i>GMAC_STATUS_BUFF_NOT_FOUND</i>	The buffer was not found in the Tx queue.
<i>GMAC_STATUS_SUCCESS</i>	Otherwise.

7.2.6.16 Gmac_Ip_GetCounter()

```
uint32 Gmac_Ip_GetCounter (
    uint8 Instance,
    Gmac_Ip_CounterType Counter )
```

Gets statistics from the specified counter.

Parameters

in	<i>instance</i>	Instance number
in	<i>counter</i>	The counter to be read

Returns

The value of the requested counter

7.2.6.17 Gmac_Ip_EnableMDIO()

```
void Gmac_Ip_EnableMDIO (
    uint8 Instance,
    boolean MiiPreambleDisabled,
    uint32 ModuleClk )
```

Enables the MDIO interface.

Parameters

in	<i>instance</i>	Instance number
in	<i>miiPreambleDisabled</i>	Enables/disables prepending a preamble to the MII management frame.
in	<i>clock</i>	provide module

7.2.6.18 Gmac_Ip_MDIORead()

```
Gmac_Ip_StatusType Gmac_Ip_MDIORead (
    uint8 Instance,
    uint8 PhyAddr,
    uint8 PhyReg,
    uint16 * Data,
    uint32 TimeoutMs )
```

Reads the selected register of the PHY.

Parameters

in	<i>instance</i>	Instance number
in	<i>phyAddr</i>	PHY device address
in	<i>phyReg</i>	PHY register address
out	<i>data</i>	Data read from the PHY
in	<i>timeoutMs</i>	Timeout for the read operation (in milliseconds)

Return values

<i>GMAC_STATUS_SUCCESS</i>	The operation completed successfully.
<i>GMAC_STATUS_TIMEOUT</i>	The specified timeout expired before completing the operation.

7.2.6.19 Gmac_Ip_MDIOWrite()

```
Gmac_Ip_StatusType Gmac_Ip_MDIOWrite (
    uint8 Instance,
    uint8 PhyAddr,
    uint8 PhyReg,
    uint16 Data,
    uint32 TimeoutMs )
```

Writes the selected register of the PHY.

Parameters

in	<i>instance</i>	Instance number
in	<i>phyAddr</i>	PHY device address
in	<i>phyReg</i>	PHY register address
in	<i>data</i>	Data to be written in the specified register of the PHY
in	<i>timeoutMs</i>	Timeout for the write operation (in milliseconds)

Return values

<i>GMAC_STATUS_SUCCESS</i>	The operation completed successfully.
<i>GMAC_STATUS_TIMEOUT</i>	The specified timeout expired before completing the operation.

7.2.6.20 Gmac_Ip_MDIOReadMMD()

```
Gmac_Ip_StatusType Gmac_Ip_MDIOReadMMD (
    uint8 Instance,
    uint8 PhyAddr,
    uint8 Mmd,
    uint16 PhyReg,
    uint16 * Data,
    uint32 TimeoutMs )
```

Reads a register of the specified MMD in a PHY device.

Parameters

in	<i>instance</i>	Instance number
in	<i>phyAddr</i>	PHY device address
in	<i>mmd</i>	The MMD index of the target register
in	<i>phyReg</i>	PHY register address
out	<i>data</i>	Data read from the PHY
in	<i>timeoutMs</i>	Timeout for the read operation (in milliseconds)

Return values

<i>GMAC_STATUS_SUCCESS</i>	The operation completed successfully.
<i>GMAC_STATUS_TIMEOUT</i>	The specified timeout expired before completing the operation.

7.2.6.21 Gmac_Ip_MDIOWriteMMD()

```
Gmac_Ip_StatusType Gmac_Ip_MDIOWriteMMD (
    uint8 Instance,
    uint8 PhyAddr,
    uint8 Mmd,
    uint16 PhyReg,
    uint16 Data,
    uint32 TimeoutMs )
```

Writes a register of the specified MMD in a PHY device.

Parameters

in	<i>instance</i>	Instance number
in	<i>phyAddr</i>	PHY device address
in	<i>mmd</i>	The MMD index of the target register
in	<i>phyReg</i>	PHY register address
in	<i>data</i>	Data to be written in the specified register of the PHY
in	<i>timeoutMs</i>	Timeout for the write operation (in milliseconds)

Return values

<i>GMAC_STATUS_SUCCESS</i>	The operation completed successfully.
<i>GMAC_STATUS_TIMEOUT</i>	The specified timeout expired before completing the operation.

7.2.6.22 Gmac_Ip_SetMacAddr()

```
void Gmac_Ip_SetMacAddr (
    uint8 Instance,
    const uint8 * MacAddr )
```

Configures the physical address of the MAC.

Parameters

in	<i>instance</i>	Instance number
in	<i>macAddr</i>	The MAC address to be configured

7.2.6.23 Gmac_Ip_GetMacAddr()

```
void Gmac_Ip_GetMacAddr (
    uint8 Instance,
    uint8 * MacAddr )
```

Gets the physical address of the MAC.

Parameters

in	<i>instance</i>	Instance number
out	<i>macAddr</i>	The physical address of the MAC

7.2.6.24 Gmac_Ip_TxTimeAwareShaperInit()

```
Gmac_Ip_StatusType Gmac_Ip_TxTimeAwareShaperInit (
    uint8 Instance,
    const Gmac_CtrlConfigType * Config )
```

Initialize time aware shaper.

Parameters

in	<i>instance</i>	Instance number
in	<i>Parameters</i>	of controller will be used. return GMAC_STATUS_SUCCESS write/read to Gate control registers successfully return GMAC_STATUS_TIMEOUT write/read to Gate control registers failed

7.2.6.25 Gmac_Ip_GetInterruptFlags()

```
uint32 Gmac_Ip_GetInterruptFlags (
    uint8 Instance )
```

Gets a mask of the common interrupt events which occurred.

Parameters

in	<i>instance</i>	Instance number
----	-----------------	-----------------

Returns

A logical OR of "Gmac_Ip_InterruptType"

7.2.6.26 Gmac_Ip_GetChInterruptFlags()

```
uint32 Gmac_Ip_GetChInterruptFlags (
    uint8 Instance,
    uint8 Channel )
```

Gets a mask of the interrupt events associated to a channel which occurred.

Parameters

in	<i>instance</i>	Instance number
in	<i>channel</i>	Channel number

Returns

A logical OR of "Gmac_Ip_ChInterruptType"

7.2.6.27 Gmac_Ip_SetBroadcastForwardAll()

```
void Gmac_Ip_SetBroadcastForwardAll (
    uint8 Instance,
    boolean Enable )
```

Enables/Disables forwarding of the broadcast traffic.

Parameters

in	<i>instance</i>	Instance number
in	<i>enable</i>	If true, the application will receive all the broadcast traffic; if false, it stops forwarding this kind of traffic.

7.2.6.28 Gmac_Ip_SetMulticastForwardAll()

```
void Gmac_Ip_SetMulticastForwardAll (
    uint8 Instance,
    boolean Enable )
```

Enables/Disables forwarding of the multicast traffic, irrespective of the destination MAC address.

Parameters

in	<i>instance</i>	Instance number
in	<i>enable</i>	If true, the application will receive all the multicast traffic; if false, stops forwarding this kind of traffic.

7.2.6.29 Gmac_Ip_SetUnicastHashFilter()

```
void Gmac_Ip_SetUnicastHashFilter (
    uint8 Instance,
    boolean Enable )
```

Enables/Disables hash filtering for unicast traffic.

Note: When the hash filter is disabled, the addresses are passed through the perfect filter. By default, the perfect filter contains the hardware address configured when the MAC module is initialized. This address is always used for destination address filtering. Optionally, a second filter can be enabled by the means of [Gmac_Ip_SetAddrPerfectFilter\(\)](#) function.

Parameters

in	<i>instance</i>	Instance number
in	<i>enable</i>	If true, the hash filtering for unicast destination addresses is enabled; if false, the hash filtering for such traffic is disabled.

7.2.6.30 Gmac_Ip_SetMulticastHashFilter()

```
void Gmac_Ip_SetMulticastHashFilter (
    uint8 Instance,
    boolean Enable )
```

Enables/Disables hash filtering for multicast traffic.

Note: When the hash filter is disabled, the addresses are passed through the perfect filter. By default, the perfect filter contains the hardware address configured when the MAC module is initialized. This address is always used for destination address filtering. Optionally, a second filter can be enabled by the means of [Gmac_Ip_SetAddrPerfectFilter\(\)](#) function.

Parameters

in	<i>instance</i>	Instance number
in	<i>enable</i>	If true, the hash filtering for multicast destination addresses is enabled; if false, the hash filtering for such traffic is disabled.

7.2.6.31 Gmac_Ip_SetHashOrPerfectFilter()

```
void Gmac_Ip_SetHashOrPerfectFilter (
    uint8 Instance,
    boolean Enable )
```

Enables/Disables either hash or perfect filters. If the destination address of an incoming packet passes either the perfect filter or the hash filter, the packet is forwarded to the application.

Note: When the hash filter is disabled, the addresses are passed through the perfect filter. By default, the perfect filter contains the hardware address configured when the MAC module is initialized. This address is always used for destination address filtering. Optionally, a second filter can be enabled by the means of [Gmac_Ip_SetAddrPerfectFilter\(\)](#) function.

Parameters

in	<i>instance</i>	Instance number
in	<i>enable</i>	If true, the hash filtering and perfect filtering for destination addresses are enabled; if false, only one filtering option is enabled, depending on unicast/multicast hash filter settings.

7.2.6.32 Gmac_Ip_AddDstAddrToHashFilter()

```
void Gmac_Ip_AddDstAddrToHashFilter (
    uint8 Instance,
    const uint8 * MacAddr )
```

Adds a hardware address to the hash filter. The destination address of an incoming packet is passed through CRC logic and then compared to the entries in the hash table.

Parameters

in	<i>instance</i>	Instance number
in	<i>macAddr</i>	The physical address

7.2.6.33 Gmac_Ip_RemoveDstAddrFromHashFilter()

```
void Gmac_Ip_RemoveDstAddrFromHashFilter (
    uint8 Instance,
    const uint8 * MacAddr )
```

Removes a hardware address from the hash filter. The destination address of an incoming packet is passed through CRC logic and then compared to the entries in the hash table.

Parameters

in	<i>instance</i>	Instance number
in	<i>macAddr</i>	The physical address

7.2.6.34 Gmac_Ip_SetAddrPerfectFilter()

```
void Gmac_Ip_SetAddrPerfectFilter (
    uint8 Instance,
    const uint8 * MacAddr,
    const uint8 Mask,
    const Gmac_Ip_MacAddrFilterType AddrType,
    boolean Enable )
```

Sets the second destination/source address perfect filter.

Note: By default, the perfect filter contains the hardware address configured when the MAC module is initialized. This address is always used for destination address filtering. Optionally, a second filter can be enabled by the means of this function. This filter allows the selection of destination or source address checking and group filtering. Only one address can be configured and it's respective mask for group filtering. The mask is 6 bits long and is used for comparing each of the MAC Address bytes. When set high, the MAC does not compare the corresponding byte of received destination address or source address with the configured MAC address. You can filter a group of addresses by masking one or more bytes of the address. When configuring the source address inverse filter, all the traffic that matches the configured source address will be rejected and the traffic that does not match the configured source address is accepted. The source address filter (either normal or inverse) result and the destination address filter result are AND'ed to decide whether the packet to be accepted. This means that the packet is dropped if either filter fails. The packet is accepted only if the packet passes both filters.

Parameters

in	<i>instance</i>	Instance number
in	<i>macAddr</i>	The physical address
in	<i>mask</i>	Group filter mask
in	<i>addrType</i>	Selects between destination address filter, source address filter or source address inverse filter
in	<i>enable</i>	If true, the application will receive all the traffic matching the selected filter; if false, it stops forwarding this kind of traffic.

7.2.6.35 Gmac_Ip_EnableVlan()

```
void Gmac_Ip_EnableVlan (
    uint8 Instance,
    const Gmac_Ip_VlanConfigType * VlanConfig )
```

Enable and set VLAN control for transmitted and received packets.

Parameters

in	<i>instance</i>	Instance number
in	<i>vlanConfig</i>	VLAN configuration structure

7.2.6.36 Gmac_Ip_SetTxOuterVlanTagForInsertion()

```
Gmac_Ip_StatusType Gmac_Ip_SetTxOuterVlanTagForInsertion (
    uint8 Instance,
    uint8 Ring,
    Gmac_Ip_VlanType VlanType,
    uint16 VlanTag )
```

Sets outer VLAN type and tag to be inserted by a particular transmission ring.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	Ring number
in	<i>vlanType</i>	VLAN type (S-VLAN or C-VLAN) to be used
in	<i>vlanTag</i>	Value for the VLAN tag to be used

Return values

<i>GMAC_STATUS_SUCCESS</i>	The ring was successfully configured.
<i>GMAC_STATUS_BUSY</i>	The resource is busy and the ring could not be configured.
<i>GMAC_STATUS_TIMEOUT</i>	The ring configuration could not complete before expiration of timeout.
<i>GMAC_STATUS_UNSUPPORTED</i>	The function was called when VLAN replacement option was configured.

7.2.6.37 Gmac_Ip_SetTxOuterVlanTagForReplacement()

```
Gmac_Ip_StatusType Gmac_Ip_SetTxOuterVlanTagForReplacement (
    uint8 Instance,
    Gmac_Ip_VlanType VlanType,
    uint16 VlanTag )
```

Sets outer VLAN tag to be replaced for all transmission rings.

Parameters

in	<i>instance</i>	Instance number
in	<i>vlanType</i>	VLAN type (S-VLAN or C-VLAN) to be used
in	<i>vlanTag</i>	Value for the VLAN tag to be used

Return values

<i>GMAC_STATUS_SUCCESS</i>	The rings were successfully configured.
<i>GMAC_STATUS_UNSUPPORTED</i>	The function was called when VLAN insertion option was configured.

7.2.6.38 Gmac_Ip_SetTxInnerVlanTag()

```
void Gmac_Ip_SetTxInnerVlanTag (
    uint8 Instance,
    Gmac_Ip_VlanType VlanType,
    uint16 VlanTag )
```

Sets inner VLAN type and tag to be inserted/replaced on transmission.

Parameters

in	<i>instance</i>	Instance number
in	<i>vlanType</i>	VLAN type (S-VLAN or C-VLAN) to be used
in	<i>vlanTag</i>	Value for the VLAN tag to be used

7.2.6.39 Gmac_Ip_AddVlanTagToHashTable()

```
void Gmac_Ip_AddVlanTagToHashTable (
    uint8 Instance,
    uint16 VlanTag )
```

Adds a VLAN Tag to the Hash Table filter.

Parameters

in	<i>instance</i>	Instance number
in	<i>vlanTag</i>	Vlan Tag to be added to the hash table

7.2.6.40 Gmac_Ip_RemoveVlanTagFromHashTable()

```
void Gmac_Ip_RemoveVlanTagFromHashTable (
    uint8 Instance,
    uint16 VlanTag )
```

Removes a VLAN Tag from the hash table filter.

Parameters

in	<i>instance</i>	Instance number
in	<i>vlanTag</i>	Vlan Tag to be removed from the hash table

7.2.6.41 Gmac_Ip_InitSysTime()

```
void Gmac_Ip_InitSysTime (
    uint8 Instance,
    const Gmac_Ip_SysTimeConfigType * SysTimeConfig )
```

Initialize system time.

Parameters

in	<i>instance</i>	Instance number
in	<i>sysTimeConfig</i>	Pointer to a structure representing the configuration of the system time

7.2.6.42 Gmac_Ip_SetSysTimeCorr()

```
Gmac_Ip_StatusType Gmac_Ip_SetSysTimeCorr (
    uint8 Instance,
    Gmac_Ip_SysTimeCorrOffsetType Offset,
    uint32 SecondsUpdate,
    uint32 NanoSecondsUpdate )
```

Set system time correction.

Parameters

in	<i>instance</i>	Instance number
in	<i>offset</i>	Chooses between negative and positive correction
in	<i>secondsUpdate</i>	The seconds part of the update
in	<i>nanosecondsUpdate</i>	The nanoseconds part of the update

Return values

<i>GMAC_STATUS_SUCCESS</i>	The correction was set with no error.
<i>GMAC_STATUS_TIMEOUT</i>	The correction could not be set before expiration of timeout.

7.2.6.43 Gmac_Ip_SetRateRatioCorr()

```
Gmac_Ip_StatusType Gmac_Ip_SetRateRatioCorr (
    uint8 Instance,
    float64 RateRatio )
```

Set system time correction.

Parameters

in	<i>instance</i>	Instance number
in	<i>rateRatio</i>	Rate ratio is used to sync

Return values

<i>GMAC_STATUS_SUCCESS</i>	The correction was set with no error.
<i>GMAC_STATUS_TIMEOUT</i>	The correction could not be set before expiration of timeout.

7.2.6.44 Gmac_Ip_GetSysTime()

```
void Gmac_Ip_GetSysTime (
    uint8 Instance,
    Gmac_Ip_TimestampType * Timestamp )
```

Gets the current system time.

Parameters

in	<i>instance</i>	Instance number
out	<i>timestamp</i>	Pointer to a structure representing the current system time

7.2.6.45 Gmac_Ip_SetTxSchedAlgo()

```
void Gmac_Ip_SetTxSchedAlgo (
    uint8 Instance,
    Gmac_Ip_TxSchedAlgoType SchedAlgo )
```

Sets the transmitter scheduling algorithm.

Parameters

in	<i>instance</i>	Instance number
in	<i>scheme</i>	Transmitter scheduling algorithm.

7.2.6.46 Gmac_Ip_SetTxRingWeight()

```
void Gmac_Ip_SetTxRingWeight (
    uint8 Instance,
    uint8 Ring,
    uint32 Weight )
```

Sets the weight (in WRR scheduling algorithm) for a particular Tx ring.

Note: If multiple rings are configured and the WRR scheduling algorithm is selected, the weight must be programmed with a non-zero value. The maximum value for the weight is 0x64.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	Tx ring
in	<i>weight</i>	Tx ring weight

7.2.6.47 Gmac_Ip_EnableTxStoreAndForward()

```
void Gmac_Ip_EnableTxStoreAndForward (
    uint8 Instance,
    uint8 Ring )
```

Enables the store and forward feature on the transmit path.

When the store and forward feature is enabled, the MAC starts to transmit data only when a complete packet is stored in the Transmit Queue.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	Tx ring

7.2.6.48 Gmac_Ip_SetTxThreshold()

```
void Gmac_Ip_SetTxThreshold (
    uint8 Instance,
    uint8 Ring,
    Gmac_Ip_TxThresholdType ThresholdValue )
```

Sets transmit threshold levels.

Parameters

in	<i>instance</i>	Instance number
in	<i>ring</i>	Tx ring
in	<i>thresholdValue</i>	The threshold value

7.3 Ethernet Driver

7.3.1 Detailed Description

Macros

- `#define ETH_43_GMAC_E_COMMUNICATION`
Runtime Error ID for "Failure or incorrect communication with the Ethernet Controller".

Function Reference

- void `Eth_43_GMAC_Init` (const `Eth_43_GMAC_ConfigType` *CfgPtr)
Initializes the Ethernet Driver.
- Std_ReturnType `Eth_43_GMAC_SetControllerMode` (uint8 CtrlIdx, `Eth_ModeType` CtrlMode)
Enables or disables the given controller.
- Std_ReturnType `Eth_43_GMAC_GetControllerMode` (uint8 CtrlIdx, `Eth_ModeType` *CtrlModePtr)
Obtains the mode of the given controller.
- Std_ReturnType `Eth_43_GMAC_TxTimeAwareShaperInit` (uint8 CtrlIdx)
Control transmit frame in duration time.
- void `Eth_43_GMAC_GetPhysAddr` (uint8 CtrlIdx, uint8 *PhysAddrPtr)
Obtains the physical source address used by the indexed controller (the node MAC address).
- void `Eth_43_GMAC_SetPhysAddr` (uint8 CtrlIdx, const uint8 *PhysAddrPtr)
Set or change physical address to the defined controller.
- BufReq_ReturnType `Eth_43_GMAC_ProvideTxBuffer` (uint8 CtrlIdx, uint8 Priority, `Eth_BufIdxType` *BufIdxPtr, uint8 **BufPtr, uint16 *LenBytePtr)
Provides access to a transmit buffer of the specified controller.
- Std_ReturnType `Eth_43_GMAC_Transmit` (uint8 CtrlIdx, `Eth_BufIdxType` BufIdx, `Eth_FrameType` FrameType, boolean TxConfirmation, uint16 LenByte, const uint8 *PhysAddrPtr)
Triggers transmission of a previously granted and then filled transmit buffer.
- void `Eth_43_GMAC_Receive` (uint8 CtrlIdx, uint8 FifoIdx, `Eth_RxStatusType` *RxStatusPtr)
Triggers frames reception notifications.
- void `Eth_43_GMAC_TxConfirmation` (uint8 CtrlIdx)
Triggers frame transmission confirmations.

7.3.2 Macro Definition Documentation

7.3.2.1 ETH_43_GMAC_E_COMMUNICATION

```
#define ETH_43_GMAC_E_COMMUNICATION
```

Runtime Error ID for "Failure or incorrect communication with the Ethernet Controller".

Definition at line 218 of file `Eth_43_GMAC.h`.

7.3.3 Function Reference

7.3.3.1 Eth_43_GMAC_Init()

```
void Eth_43_GMAC_Init (
    const Eth_43_GMAC_ConfigType * CfgPtr )
```

Initializes the Ethernet Driver.

The configuration pointer is internally stored and the driver is initialized. The Ethernet controller is also reset.

Note

Function should be called only once.

Warning

Second call can cause undefined behavior. Call the Eth_SetControllerMode() and pass ETH_MODE_DOWN to the CtrlMode argument before the second Eth_Init call to avoid problems.

Parameters

in	CfgPtr	Points to the implementation specific structure containing the Eth driver configuration Compiler_Warning: this warning due to behavior of compiler depend on configs. implements Eth_Init_Activity
----	--------	---

7.3.3.2 Eth_43_GMAC_SetControllerMode()

```
Std_ReturnType Eth_43_GMAC_SetControllerMode (
    uint8 CtrlIdx,
    Eth_ModeType CtrlMode )
```

Enables or disables the given controller.

Warning

Disabling the controller clears all receive and transmit buffers. The application should ensure that no data is lost.

Parameters

in	CtrlIdx	Index of the controller to be enabled or disabled. The index is valid within the context of the Ethernet Driver only.
in	CtrlMode	Mode which shall be entered <ul style="list-style-type: none">ETH_MODE_DOWN: disable the controller
178		<ul style="list-style-type: none">ETH_MODE_ACTIVE: enable the controller

Returns

Error status

Return values

<i>E_OK</i>	No error was detected during the function execution.
<i>E_NOT_OK</i>	Development error was detected and the function failed. implements Eth_SetControllerMode_Activity

7.3.3.3 Eth_43_GMAC_GetControllerMode()

```
Std_ReturnType Eth_43_GMAC_GetControllerMode (
    uint8 CtrlIdx,
    Eth_ModeType * CtrlModePtr )
```

Obtains the mode of the given controller.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which state shall be read. The index is valid within the context of the Ethernet Driver only.
out	<i>CtrlModePtr</i>	Pointer where to store the current controller mode.

Returns

Error status

Return values

<i>E_OK</i>	No error was detected during the function execution.
<i>E_NOT_OK</i>	Development error was detected and the function failed. implements Eth_GetControllerMode_Activity

7.3.3.4 Eth_43_GMAC_TxTimeAwareShaperInit()

```
Std_ReturnType Eth_43_GMAC_TxTimeAwareShaperInit (
    uint8 CtrlIdx )
```

Control transmit frame in duration time.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which state shall be read. The index is valid within the context of the Ethernet Driver only.
----	----------------	---

Returns

Std_ReturnType

7.3.3.5 Eth_43_GMAC_GetPhysAddr()

```
void Eth_43_GMAC_GetPhysAddr (
    uint8 CtrlIdx,
    uint8 * PhysAddrPtr )
```

Obtains the physical source address used by the indexed controller (the node MAC address).

Parameters

in	<i>CtrlIdx</i>	Index of the controller whose MAC address should be read. The index is valid within the context of the Ethernet Driver only.
out	<i>PhysAddrPtr</i>	Pointer where to store the physical source address (MAC address). The address in network byte order is stored into 6 bytes at the given memory address. implements Eth_GetPhysAddr_Activity

7.3.3.6 Eth_43_GMAC_SetPhysAddr()

```
void Eth_43_GMAC_SetPhysAddr (
    uint8 CtrlIdx,
    const uint8 * PhysAddrPtr )
```

Set or change physical address to the defined controller.

Parameters

in	<i>CtrlIdx</i>	Index of the controller whose MAC address should be changed. The index is valid within the context of the Ethernet Driver only.
in	<i>PhysAddrPtr</i>	Pointer to the physical source address which should be set to the controller. The address is stored in 6 bytes of memory in network byte order.

Warning

This function may be called only when the controller is down. Calling function `Eth_Init` will change the controller's MAC address to the default value! implements `Eth_SetPhysAddr_Activity`

7.3.3.7 Eth_43_GMAC_ProvideTxBuffer()

```
BufReq_ReturnType Eth_43_GMAC_ProvideTxBuffer (
    uint8 CtrlIdx,
    uint8 Priority,
    Eth_BufIdxType * BufIdxPtr,
    uint8 ** BufPtr,
    uint16 * LenBytePtr )
```

Provides access to a transmit buffer of the specified controller.

Warning

The application should handle possible difference between the requested and granted buffer lengths. It is not necessary to use whole granted buffer i.e. some space at the end may not be written.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which buffer shall be provided. The index is valid within the context of the Ethernet Driver only.
in	<i>Priority</i>	Frame priority for transmit buffer FIFO selection
out	<i>BufIdxPtr</i>	Index to the granted transmit buffer resource. It uniquely identifies the buffer in all subsequent calls of functions <code>Eth_Transmit()</code> and <code>Eth_TxConfirmation()</code> .
out	<i>BufPtr</i>	Pointer to the granted buffer. This is the space where the data to be transmitted shall be stored.
in, out	<i>LenBytePtr</i>	Buffer payload length <ul style="list-style-type: none"> • In: desired length in bytes • Out: granted length in bytes

Returns

Error and buffer status

Return values

<i>BUFREQ_OK</i>	Buffer was successfully granted and no error has occurred.
<i>BUFREQ_E_NOT_OK</i>	A development error was detected and no buffer was granted.
<i>BUFREQ_E_BUSY</i>	All available buffers in use therefore no buffer was granted. No error has been detected. implements <code>Eth_ProvideTxBuffer_Activity</code>

7.3.3.8 Eth_43_GMAC_Transmit()

```
Std_ReturnType Eth_43_GMAC_Transmit (
    uint8 CtrlIdx,
    Eth_BufIdxType BufIdx,
    Eth_FrameType FrameType,
    boolean TxConfirmation,
    uint16 LenByte,
    const uint8 * PhysAddrPtr )
```

Triggers transmission of a previously granted and then filled transmit buffer.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which buffer shall be transmitted. The index is valid within the context of the Ethernet Driver only.
in	<i>BufIdx</i>	Index of the buffer resource to be transmitted.
in	<i>FrameType</i>	Desired value of the Ethernet frame type in the frame header.
in	<i>TxConfirmation</i>	Activates transmission confirmation.
in	<i>LenByte</i>	Buffer data length in bytes (payload length).
in	<i>PhysAddrPtr</i>	Physical target address (MAC address) in network byte order.

Returns

Error status

Return values

<i>E_OK</i>	No error was detected during the function execution.
<i>E_NOT_OK</i>	Development error was detected and the function failed. implements Eth_Transmit_Activity

7.3.3.9 Eth_43_GMAC_Receive()

```
void Eth_43_GMAC_Receive (
    uint8 CtrlIdx,
    uint8 FifoIdx,
    Eth_RxStatusType * RxStatusPtr )
```

Triggers frames reception notifications.

All receive buffers are checked and the first received frame is passed to the EthIf module. The caller is notified whether any frame was received and whether more frames are available in the receive queue.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which shall be checked whether any new frames were received. The index is valid within the context of the Ethernet Driver only.
in	<i>FifoIdx</i>	Specifies the related fifo
out	<i>RxStatusPtr</i>	Inform the caller whether a frame was received (ETH_RECEIVED or ETH_NOT_RECEIVED) and whether more frames are available in the queue (ETH_RECEIVED or ETH_RECEIVED_MORE_DATA_AVAILABLE). implements Eth_Receive_Activity

7.3.3.10 Eth_43_GMAC_TxConfirmation()

```
void Eth_43_GMAC_TxConfirmation (
    uint8 CtrlIdx )
```

Triggers frame transmission confirmations.

Parameters

in	<i>CtrlIdx</i>	Index of the controller which shall be checked whether any frame transmission has finished. The index is valid within the context of the Ethernet Driver only.
----	----------------	--

All transmit buffers are checked and upper layers are informed about successfully transmitted frames. Buffers containing transmitted frames are unlocked after the confirmation. implements Eth_TxConfirmation_Activity



Chapter 8

Data Structure Documentation

8.1 Gmac_Ip_ManagementInfo Struct Reference

Management Frame information.

```
#include <Gmac_Ip_Hw_Access.h>
```

8.1.1 Detailed Description

Management Frame information.

Definition at line 109 of file Gmac_Ip_Hw_Access.h.

How to Reach Us:

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, and UMEMS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. ARM, AMBA, ARM Powered, Artisan, Cortex, Jazelle, Keil, SecurCore, Thumb, TrustZone, and Vision are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. ARM7, ARM9, ARM11, big.LITTLE, CoreLink, CoreSight, DesignStart, Mali, mbed, NEON, POP, Sensinode, Socrates, ULINK and Versatile are trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2023 NXP B.V.

