

MCAL Configuration Verification Manual for Eth_17_GEthMac

32-bit TriCore™ AURIX™ TC3xx microcontroller family

About this document

Scope and purpose

This Configuration Data Reference document is applicable to all TC3xx devices in the TriCore™ AURIX™ family of 32-bit microcontrollers.

The purpose of this document is to facilitate the integrator to verify the generated code based on the input configuration parameters. This document describes details of structures, defines, macros and variables generated from the configuration parameters.

Intended audience

This document is intended for integrators who need to understand the logic of the generated configuration code of AURIX™ AUTOSAR MCAL.

Reference documents

This document should be read in conjunction with the following documents:

- AURIX™ TC3XX MCAL User Manual Eth_17_GEthMac

Table of contents

About this document.....	1
Table of contents.....	2
1 Eth_17_GEthMac driver.....	4
1.1 File: Eth_17_GEthMac_Cfg.h.....	4
1.1.1 Macro: ETH_17_GETHMAC_AR_RELEASE_MAJOR_VERSION	4
1.1.2 Macro: ETH_17_GETHMAC_AR_RELEASE_MINOR_VERSION	4
1.1.3 Macro: ETH_17_GETHMAC_AR_RELEASE_REVISION_VERSION	4
1.1.4 Macro: ETH_17_GETHMAC_SW_MAJOR_VERSION	5
1.1.5 Macro: ETH_17_GETHMAC_SW_MINOR_VERSION	5
1.1.6 Macro: ETH_17_GETHMAC_SW_PATCH_VERSION	5
1.1.7 Macro: ETH_17_GETHMAC_GETDROPCOUNT_API.....	6
1.1.8 Macro: ETH_17_GETHMAC_GETETHERSTATS_API.....	6
1.1.9 Macro: ETH_17_GETHMAC_ENA_MII_API.....	6
1.1.10 Macro: ETH_17_GETHMAC_DEM_ENABLED.....	7
1.1.11 Macro: ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API	7
1.1.12 Macro: ETH_17_GETHMAC_GLOBALTIMESUPPORT	7
1.1.13 Macro: ETH_17_GETHMAC_MULTICORE_ERROR_DETECT	8
1.1.14 Macro: ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE	8
1.1.15 Macro: ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE	8
1.1.16 Macro: ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE	9
1.1.17 Macro: ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE.....	9
1.1.18 Macro: ETH_17_GETHMAC_RUNTIME_API_MODE.....	10
1.1.19 Macro: ETH_17_GETHMAC_INIT_API_MODE.....	10
1.1.20 Macro: ETH_17_GETHMAC_DEV_ERROR_DETECT.....	10
1.1.21 Macro: ETH_17_GETHMAC_FGETH_IN_HZ	11
1.1.22 Macro: ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC	11
1.1.23 Macro: ETH_17_GETHMAC_INDEX.....	11
1.1.24 Macro: ETH_17_GETHMAC_MAXTIMEOUT_COUNT	12
1.1.25 Macro: ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL	12
1.1.26 Macro: ETH_17_GETHMAC_VERSION_INFO_API	12
1.1.27 Macro: ETH_17_GETHMAC_MAX_CORES	13
1.1.28 Macro: ETH_17_GETHMAC_MAX_CONTROLLERS.....	13
1.1.29 Macro: Eth_17_GEthMacConf_EthCtrlConfig_<Container Name>.....	13
1.1.30 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_COUNT	14
1.1.31 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_SIZE	14
1.1.32 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_TXBUFFER_SIZE	15
1.1.33 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_TXBUFFER_COUNT	16
1.1.34 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_CORE<Core Id>.....	16
1.1.35 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_CONFIGURED.....	17
1.1.36 Macro: ETH_17_GETHMAC_MAX_CNTRL_CORE<Core Id>.....	17
1.1.37 Macro: ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT	17
1.1.38 Macro: ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT	18
1.1.39 Macro: ETH_17_GETHMAC_DMA_RESET_WAITCYCLE.....	18
1.2 File: Eth_17_GEthMac[_<variant>]_PBCfg.c.....	18
1.2.1 Structure: Eth_17_GEthMac_CoreCntrlConfigCore<Core Id>[_Variant][Max controllers Configured].....	19
1.2.1.1 Member: EthSkewDelay.....	23

Table of contents

1.2.1.2	Member: EthGptclRegVal	24
1.2.1.3	Member: EthCtrlRxBufLenByte	24
1.2.1.4	Member: EthCtrlRxBufLenByteAligned	25
1.2.1.5	Member: EthCtrlTxBufLenByte	26
1.2.1.6	Member: EthCtrlTxBufLenByteAligned	26
1.2.1.7	Member: EthCntrlProperties	27
1.2.1.8	Member: EthDemAccess	27
1.2.1.9	Member: EthDemFramesLost	28
1.2.1.10	Member: EthDemAlignment	28
1.2.1.11	Member: EthDemCRC	29
1.2.1.12	Member: EthDemUndersize	29
1.2.1.13	Member: EthDemOversize	30
1.2.1.14	Member: EthDemSingleCollision	30
1.2.1.15	Member: EthDemMultipleCollision	31
1.2.1.16	Member: EthDemLateCollision	31
1.2.1.17	Member: EthRxBufTotal	32
1.2.1.18	Member: EthTxBufTotal	32
1.2.1.19	Member: EthCntrlIdx	32
1.2.1.20	Member: EthMacAddress [6]	33
1.2.2	Structure: Eth_17_GEthMac_ConfigCore<Core Id>[_Variant]	33
1.2.2.1	Member: EthCoreCntrlPtr	34
1.2.2.2	Member: EthMaxControllers	34
1.2.3	Structure: Eth_17_GEthMac_Config[_Variant]	35
1.2.3.1	Member: EthCoreAdd[ETH_17_GETHMAC_MAX_CORES]	36
1.2.3.2	Member: EthNodeIdxmapPtr	37
1.2.4	Array: Eth_17_GEthMac_ControllerIndexMap[_Variant][Max controllers Configured]	37
1.3	File: Eth_17_GEthmac[_<variant>]_PBcfg.h	38
1.3.1	Structure: Eth_17_GEthMac_Config[_<variant>]	38
Revision history		39

1 Eth_17_GEthMac driver

This chapter describes the details of the configuration data generated from the ETH driver.

1.1 File: Eth_17_GEthMac_Cfg.h

The generated header file contains all pre-compile configuration parameters. Pre-compile time configuration allows decoupling of the static configuration from implementation. The file is generated in 'inc' folder.

1.1.1 Macro: ETH_17_GETHMAC_AR_RELEASE_MAJOR_VERSION

Table 1 ETH_17_GETHMAC_AR_RELEASE_MAJOR_VERSION

Name	ETH_17_GETHMAC_AR_RELEASE_MAJOR_VERSION	
Description	Major version number of AUTOSAR release on which the Eth_17_GEthMac implementation is based on.	
Verification method	The macro is generated with the value present in 'CommonPublishedInformation/ArMajorVersion'. <i>Note: The macro is not user configurable.</i>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with ArMajorVersion 4	#define ETH_17_GETHMAC_AR_RELEASE_MAJOR_VERSION (4U)

1.1.2 Macro: ETH_17_GETHMAC_AR_RELEASE_MINOR_VERSION

Table 2 ETH_17_GETHMAC_AR_RELEASE_MINOR_VERSION

Name	ETH_17_GETHMAC_AR_RELEASE_MINOR_VERSION	
Description	Minor version number of AUTOSAR release on which the Eth_17_GEthMac implementation is based on.	
Verification method	The macro is generated with the value present in 'CommonPublishedInformation/ArMinorVersion'. <i>Note: The macro is not user configurable.</i>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with ArMinorVersion 2	#define ETH_17_GETHMAC_AR_RELEASE_MINOR_VERSION (2U)

1.1.3 Macro: ETH_17_GETHMAC_AR_RELEASE_REVISION_VERSION

Table 3 ETH_17_GETHMAC_AR_RELEASE_REVISION_VERSION

Eth_17_GEthMac driver

Name	ETH_17_GETHMAC_AR_RELEASE_REVISION_VERSION	
Description	Revision version number of AUTOSAR release on which the Eth_17_GEthMac implementation is based on.	
Verification method	<p>The macro is generated with the value present in 'CommonPublishedInformation/ArPatchVersion'.</p> <p><i>Note: The macro is not user configurable.</i></p>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with ArPatchVersion 2	#define ETH_17_GETHMAC_AR_RELEASE_REVISION_VERSION (2U)

1.1.4 Macro: ETH_17_GETHMAC_SW_MAJOR_VERSION**Table 4 ETH_17_GETHMAC_SW_MAJOR_VERSION**

Name	ETH_17_GETHMAC_SW_MAJOR_VERSION	
Description	Major version number of the Eth_17_GEthMac module.	
Verification method	<p>The macro is generated with the value present in 'CommonPublishedInformation/SwMajorVersion'.</p> <p><i>Note: The macro is not user configurable.</i></p>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with SwMajorVersion 10	#define ETH_17_GETHMAC_SW_MAJOR_VERSION (10U)

1.1.5 Macro: ETH_17_GETHMAC_SW_MINOR_VERSION**Table 5 ETH_17_GETHMAC_SW_MINOR_VERSION**

Name	ETH_17_GETHMAC_SW_MINOR_VERSION	
Description	Minor version number of the Eth_17_GEthMac module.	
Verification method	<p>The macro is generated with the value present in 'CommonPublishedInformation/SwMinorVersion'.</p> <p><i>Note: The macro is not user configurable.</i></p>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with SwMinorVersion 30	#define ETH_17_GETHMAC_SW_MINOR_VERSION (30U)

1.1.6 Macro: ETH_17_GETHMAC_SW_PATCH_VERSION**Table 6 ETH_17_GETHMAC_SW_PATCH_VERSION**

Name	ETH_17_GETHMAC_SW_PATCH_VERSION	
Description	Patch level version number of the Eth_17_GEthMac module.	

Eth_17_GEthMac driver

Verification method	The macro is generated with the value present in 'CommonPublishedInformation/SwPatchVersion'.	
	<i>Note: The macro is not user configurable.</i>	
Example(s)	Action	Generated output
	Generate Eth_17_GEthMac_Cfg.h file with SwPatchVersion 0	#define ETH_17_GETHMAC_SW_PATCH_VERSION (0U)

1.1.7 Macro: ETH_17_GETHMAC_GETDROPCOUNT_API

Table 7 ETH_17_GETHMAC_GETDROPCOUNT_API

Name	ETH_17_GETHMAC_GETDROPCOUNT_API	
Description	Enables/disables Eth_17_GEthMac_GetDropCount API	
Verification method	The macro is generated as STD_ON if EthGetDropCountApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthGetDropCountApi = True	#define ETH_17_GETHMAC_GETDROPCOUNT_API (STD_ON)
	EthGetDropCountApi = False	#define ETH_17_GETHMAC_GETDROPCOUNT_API (STD_OFF)

1.1.8 Macro: ETH_17_GETHMAC_GETETHERSTATS_API

Table 8 ETH_17_GETHMAC_GETETHERSTATS_API

Name	ETH_17_GETHMAC_GETETHERSTATS_API	
Description	Enables/disables Eth_17_GEthMac_GetEtherStats API	
Verification method	The macro is generated as STD_ON if EthGetEtherStatsApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthGetEtherStatsApi = True	#define ETH_17_GETHMAC_GETETHERSTATS_API (STD_ON)
	EthGetEtherStatsApi = False	#define ETH_17_GETHMAC_GETETHERSTATS_API (STD_OFF)

1.1.9 Macro: ETH_17_GETHMAC_ENA_MII_API

Table 9 ETH_17_GETHMAC_ENA_MII_API

Eth_17_GEthMac driver

Name	ETH_17_GETHMAC_ENA_MII_API	
Description	Enables/disables Eth_17_GEthMac_WriteMii and Eth_17_GEthMac_ReadMii APIs	
Verification method	The macro is generated as STD_ON if EthCtrlEnableMii configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthCtrlEnableMii = True	#define ETH_17_GETHMAC_ENA_MII_API (STD_ON)
	EthCtrlEnableMii = False	#define ETH_17_GETHMAC_ENA_MII_API (STD_OFF)

1.1.10 Macro: ETH_17_GETHMAC_DEM_ENABLED

Table 10 ETH_17_GETHMAC_DEM_ENABLED

Name	ETH_17_GETHMAC_DEM_ENABLED	
Description	Enables/disables DEM reporting	
Verification method	The macro is generated as ETH_17_GETHMAC_ENABLE_DEM_REPORT if node exist for any one of the EthDemEventParameterRefs else the macro is generated as ETH_17_GETHMAC_DISABLE_DEM_REPORT.	
Example(s)	Action	Generated output
	Configure node in EthConfigSet/ EthCtrlConfig/ EthCtrlConfig_0/ EthDemEventParameterRefs/ EthDemEventParameterRefs_0/ ETH_E_ACCESS	#define ETH_17_GETHMAC_DEM_ENABLED (ETH_17_GETHMAC_ENABLE_DEM_REPORT)
	No node is configured	#define ETH_17_GETHMAC_DEM_ENABLED (ETH_17_GETHMAC_DISABLE_DEM_REPORT)

1.1.11 Macro: ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API

Table 11 ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API

Name	ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API	
Description	Enables/disables Eth_17_GEthMac_UpdatePhysAddrFilter API	
Verification method	The macro is generated as STD_ON if EthUpdatePhysAddrFilter configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthUpdatePhysAddrFilter = True	#define ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API (STD_ON)
	EthUpdatePhysAddrFilter = False	#define ETH_17_GETHMAC_UPDATE_PHY_ADDR_FILTER_API (STD_OFF)

1.1.12 Macro: ETH_17_GETHMAC_GLOBALTIMESUPPORT

Table 12 ETH_17_GETHMAC_GLOBALTIMESUPPORT

Name	ETH_17_GETHMAC_GLOBALTIMESUPPORT	
Description	Enables/disables Eth_17_GEthMac_GetCurrentTime, Eth_17_GEthMac_EnableEgressTimeStamp, Eth_17_GEthMac_GetEgressTimeStamp, Eth_17_GEthMac_GetIngressTimeStamp, Eth_17_GEthMac_SetCorrectionTime, Eth_17_GEthMac_SetGlobalTime APIs	
Verification method	The macro is generated as STD_ON if EthGlobalTimeSupport configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthGlobalTimeSupport = True	#define ETH_17_GETHMAC_GLOBALTIMESUPPORT (STD_ON)
	EthGlobalTimeSupport = False	#define ETH_17_GETHMAC_GLOBALTIMESUPPORT (STD_OFF)

1.1.13 Macro: ETH_17_GETHMAC_MULTICORE_ERROR_DETECT

Table 13 ETH_17_GETHMAC_MULTICORE_ERROR_DETECT

Name	ETH_17_GETHMAC_MULTICORE_ERROR_DETECT	
Description	Enables/disables multi core error detection and reporting from the core.	
Verification method	The macro is generated as STD_ON if EthMultiCoreErrorDetect configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthMultiCoreErrorDetect = True	#define ETH_17_GETHMAC_MULTICORE_ERROR_DETECT (STD_ON)
	EthMultiCoreErrorDetect = False	#define ETH_17_GETHMAC_MULTICORE_ERROR_DETECT (STD_OFF)

1.1.14 Macro: ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE

Table 14 ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE

Name	ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE	
Description	Enables/disables offloading of ICMP frames for both transmission and reception.	
Verification method	The macro is generated as STD_ON if EthCtrlEnableOffloadChecksumICMP configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthCtrlEnableOffloadChecksumICMP = True	#define ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE (STD_ON)
	EthCtrlEnableOffloadChecksumICMP = False	#define ETH_17_GETHMAC_ICMP_CHECKSUMOFFLOAD_ENABLE (STD_OFF)

1.1.15 Macro: ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE

Table 15 ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE

Name	ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE	
Description	Enables/disables offloading of IPv4 frames for both transmission and reception.	
Verification method	The macro is generated as STD_ON if EthCtrlEnableOffloadChecksumIPv4 configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthCtrlEnableOffloadChecksumIPv4 = True	#define ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE (STD_ON)
	EthCtrlEnableOffloadChecksumIPv4 = False	#define ETH_17_GETHMAC_IPV4_CHECKSUMOFFLOAD_ENABLE (STD_OFF)

1.1.16 Macro: ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE

Table 16 ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE

Name	ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE	
Description	Enables/disables offloading of TCP frames for both transmission and reception.	
Verification method	The macro is generated as STD_ON if EthCtrlEnableOffloadChecksumTCP configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthCtrlEnableOffloadChecksumTCP = True	#define ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE (STD_ON)
	EthCtrlEnableOffloadChecksumTCP = False	#define ETH_17_GETHMAC_TCP_CHECKSUMOFFLOAD_ENABLE (STD_OFF)

1.1.17 Macro: ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE

Table 17 ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE

Name	ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE	
Description	Enables/disables offloading of UDP frames for both transmission and reception.	
Verification method	The macro is generated as STD_ON if EthCtrlEnableOffloadChecksumUDP configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthCtrlEnableOffloadChecksumUDP = True	#define ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE (STD_ON)
	EthCtrlEnableOffloadChecksumUDP = False	#define ETH_17_GETHMAC_UDP_CHECKSUMOFFLOAD_ENABLE (STD_OFF)

1.1.18 Macro: ETH_17_GETHMAC_RUNTIME_API_MODE

Table 18 ETH_17_GETHMAC_RUNTIME_API_MODE

Name	ETH_17_GETHMAC_RUNTIME_API_MODE	
Description	Decides the mode of execution of Run Time API's	
Verification method	The macro is generated as ETH_17_GETHMAC_MCAL_USER1 if EthRuntimeApiMode configuration parameter is set to 'ETH_MCAL_USER1' else the macro is generated as ETH_17_GETHMAC_MCAL_SUPERVISOR.	
Example(s)	Action	Generated output
	EthRuntimeApiMode = ETH_MCAL_USER1	#define ETH_17_GETHMAC_RUNTIME_API_MODE (ETH_17_GETHMAC_MCAL_USER1)
	EthRuntimeApiMode = ETH_MCAL_SUPERVISOR	#define ETH_17_GETHMAC_RUNTIME_API_MODE (ETH_17_GETHMAC_MCAL_SUPERVISOR)

1.1.19 Macro: ETH_17_GETHMAC_INIT_API_MODE

Table 19 ETH_17_GETHMAC_INIT_API_MODE

Name	ETH_17_GETHMAC_INIT_API_MODE	
Description	Decides the mode of execution of the Init API.	
Verification method	The macro is generated as ETH_17_GETHMAC_MCAL_USER1 if EthInitApiMode configuration parameter is set to 'ETH_MCAL_USER1' else the macro is generated as ETH_17_GETHMAC_MCAL_SUPERVISOR.	
Example(s)	Action	Generated output
	EthInitApiMode = ETH_MCAL_USER1	#define ETH_17_GETHMAC_INIT_API_MODE (ETH_17_GETHMAC_MCAL_USER1)
	EthInitApiMode = ETH_MCAL_SUPERVISOR	#define ETH_17_GETHMAC_INIT_API_MODE (ETH_17_GETHMAC_MCAL_SUPERVISOR)

1.1.20 Macro: ETH_17_GETHMAC_DEV_ERROR_DETECT

Table 20 ETH_17_GETHMAC_DEV_ERROR_DETECT

Name	ETH_17_GETHMAC_DEV_ERROR_DETECT	
Description	Enables/disables the Development Error Detection.	
Verification method	The macro is generated as STD_ON if EthDevErrorDetect configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthDevErrorDetect = True	#define ETH_17_GETHMAC_DEV_ERROR_DETECT (STD_ON)
	EthDevErrorDetect = False	#define ETH_17_GETHMAC_DEV_ERROR_DETECT (STD_OFF)

1.1.21 Macro: ETH_17_GETHMAC_FGETH_IN_HZ

Table 21 ETH_17_GETHMAC_FGETH_IN_HZ

Name	ETH_17_GETHMAC_FGETH_IN_HZ	
Description	Basic frequency for gigabit Ethernet kernel in Hertz.	
Verification method	The macro is generated as the value configured by the user in EthOperateFrequency configuration parameter which in turn refers McuGEthFrequency configuration parameter in the MCU module.	
Example(s)	Action	Generated output
	EthOperateFrequency = McuGEthFrequency = 60000000	#define ETH_17_GETHMAC_FGETH_IN_HZ (60000000U)
	EthOperateFrequency = McuGEthFrequency = 150000000	#define ETH_17_GETHMAC_FGETH_IN_HZ (150000000U)

1.1.22 Macro: ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC

Table 22 ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC

Name	ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC	
Description	SPB frequency in nanoseconds.	
Verification method	The macro is generated as the value configured by the user in EthPeripheralBusFrequency configuration parameter which in turn refers McuSPBFrequency configuration parameter in the MCU module.	
Example(s)	Action	Generated output
	EthPeripheralBusFrequency = McuSPBFrequency = 100000000 EthSpbPeriodInNanoSeconds =1000000000/EthPeripheralBusFrequency =10	#define ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC (10U)
	EthPeripheralBusFrequency = McuSPBFrequency = 300000000 EthSpbPeriodInNanoSeconds =1000000000/EthPeripheralBusFrequency =30	#define ETH_17_GETHMAC_FSPB_PERIOD_IN_NANOSEC (30U)

1.1.23 Macro: ETH_17_GETHMAC_INDEX

Table 23 ETH_17_GETHMAC_INDEX

Name	ETH_17_GETHMAC_INDEX
Description	Ethernet driver instance ID.
Verification method	The macro is generated as a numeric value set in the configuration parameter 'EthGeneral/EthIndex'

Example(s)	Action	Generated output
	Set EthIndex = 0	#define ETH_17_GETHMAC_INDEX (0U)
	Set EthIndex = 5	#define ETH_17_GETHMAC_INDEX (5U)

1.1.24 Macro: ETH_17_GETHMAC_MAXTIMEOUT_COUNT

Table 24 ETH_17_GETHMAC_MAXTIMEOUT_COUNT

Name	ETH_17_GETHMAC_MAXTIMEOUT_COUNT	
Description	Specifies maximum timeout count in nanoseconds for hardware timeout errors	
Verification method	The macro is generated as the value set in the configuration parameter 'EthGeneral/EthTimeoutCount'.	
Example(s)	Action	Generated output
	Set EthTimeoutCount as 100	#define ETH_17_GETHMAC_MAXTIMEOUT_COUNT (100U)
	Set EthTimeoutCount as 4294967295	#define ETH_17_GETHMAC_MAXTIMEOUT_COUNT (4294967295U)

1.1.25 Macro: ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL

Table 25 ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL

Name	ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL	
Description	Clock configuration for MDIO	
Verification method	The macro is generated based on EthPeripheralBusClock configuration parameter. MDIO clock is between 1.0 MHz to 2.5 MHz frequency based on SPB clock frequency.	
Example(s)	Action	Generated output
	If EthPeripheralBusClock is between ≥ 60000000 and ≤ 100000000	#define ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL (ETH_17_GETHMAC_CR60_100MHZ)
	If EthPeripheralBusClock is between > 100000000 and ≤ 150000000	#define ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL (ETH_17_GETHMAC_CR100_150MHZ)
	If EthPeripheralBusClock is between > 150000000 and ≤ 250000000	#define ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL (ETH_17_GETHMAC_CR150_250MHZ)
	If EthPeripheralBusClock is between ≥ 250000000 and ≤ 300000000	#define ETH_17_GETHMAC_MDIO_ADDR_REG_CR_VAL (ETH_17_GETHMAC_CR250_300MHZ)

1.1.26 Macro: ETH_17_GETHMAC_VERSION_INFO_API

Table 26 ETH_17_GETHMAC_VERSION_INFO_API

Name	ETH_17_GETHMAC_VERSION_INFO_API
Description	Enables/disables Eth_17_GEthMac_GetVersionInfo API

Verification method	The macro is generated as STD_ON if EthVersionInfoApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	EthVersionInfoApi = True	#define ETH_17_GETHMAC_VERSION_INFO_API (STD_ON)
	EthVersionInfoApi = False	#define ETH_17_GETHMAC_VERSION_INFO_API (STD_OFF)

1.1.27 Macro: ETH_17_GETHMAC_MAX_CORES

Table 27 ETH_17_GETHMAC_MAX_CORES

Name	ETH_17_GETHMAC_MAX_CORES	
Description	Maximum available cores in the device.	
	<i>Note: This macro is not configurable by the user.</i>	
Verification method	The macro is generated based on maximum number of cores available.	
Example(s)	Action	Generated output
	Device has 6 cores.	#define ETH_17_GETHMAC_MAX_CORES (6U)
	Device has 4 cores.	#define ETH_17_GETHMAC_MAX_CORES (4U)

1.1.28 Macro: ETH_17_GETHMAC_MAX_CONTROLLERS

Table 28 ETH_17_GETHMAC_MAX_CONTROLLERS

Name	ETH_17_GETHMAC_MAX_CONTROLLERS	
Description	Maximum available controllers in the device.	
	<i>Note: This macro is not configurable by the user.</i>	
Verification method	The macro is generated based on maximum number of controllers available.	
Example(s)	Action	Generated output
	Device has 1 controller.	#define ETH_17_GETHMAC_MAX_CONTROLLERS (1U)
	Device has 2 controllers.	#define ETH_17_GETHMAC_MAX_CONTROLLERS (2U)

1.1.29 Macro: Eth_17_GEthMacConf_EthCtrlConfig_<Container Name>

Table 29 Eth_17_GEthMacConf_EthCtrlConfig_<Container Name>

Name	Eth_17_GEthMacConf_EthCtrlConfig_<Container Name>
-------------	---

Eth_17_GEthMac driver

Description	The macro is the symbolic name generated for the configuration parameter EthCtrlIdx	
Verification method	The macro is generated as a numeric value which is configured in EthCtrlIdx. <Container Name> is the name of the Ethernet controller container.	
Example(s)	Action	Generated output
	If EthCtrlIdx is 1	#define Eth_17_GEthMacConf_EthCtrlConfig_EthCtrlConfig_0 (1U)

1.1.30 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_COUNT**Table 30 ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_COUNT**

Name	ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_COUNT	
Description	Selects the total number of receive buffers for Eth driver for the controller with <Controller Index>.	
Verification method	The macro is generated as a numeric value which is configured in EthRxBufTotal.	
Example(s)	Action	Generated output
	Set EthRxBufTotal = 0 for CtrlIdx 0.	#define ETH_17_GETHMAC_CNTRL0_RXBUFFER_COUNT (1U)
	Set EthRxBufTotal = 255 for CtrlIdx 1.	#define ETH_17_GETHMAC_CNTRL1_RXBUFFER_COUNT (255U)

1.1.31 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_SIZE**Table 31 ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_SIZE**

Name	ETH_17_GETHMAC_CNTRL<Controller Index>_RXBUFFER_SIZE	
Description	Total size of RAM allocated for receive buffers for the controller with <Controller Index>.	
Verification method	The macro is generated as total receive buffer size with 8 byte alignment (value of the macro is a product of the values provided in EthRxBufTotal and EthCtrlRxBufLenBytevalue configuration parameters).	
	<i>Note: If EthCtrlRxBufLenByte is not 8-byte aligned, then it is changed to next 8-byte aligned value. This is done to align the Rx buffers on 8-byte aligned addresses for maximum hardware performance.</i>	
Example(s)	Action	Generated output
	Set EthRxBufTotal to 255 and EthCtrlRxBufLenBytevalue to 1522 for CtrlIdx 0 <i>Note: 8 byte alignment for EthCtrlRxBufLenBytevalue : if (EthCtrlRxBufLenBytevalue mod 8 != 0) then</i>	#define ETH_17_GETHMAC_CNTRL0_RXBUFFER_SIZE (389640U)

$((1522/8)+1)*8 = 191 * 8 = 1528$	
Set EthRxBufTotal to 1 and EthCtrlRxBufLenBytevalue to 1522 for CtrlIdx 1 <i>Note:</i> 8 byte alignment for EthCtrlRxBufLenBytevalue : if (EthCtrlRxBufLenBytevalue mod 8 != 0) then $((1522/8)+1)*8 = 191 * 8 = 1528$	#define ETH_17_GETHMAC_CNTRL1_RXBUFFER_SIZE (1528U)
Set EthRxBufTotal to 4 and EthCtrlRxBufLenBytevalue to 0 for CtrlIdx 1 <i>Note:</i> 8 byte alignment for EthCtrlRxBufLenBytevalue : if (EthCtrlRxBufLenBytevalue mod 8 != 0) then $((1522/8)+1)*8 = 191 * 8 = 1528$	#define ETH_17_GETHMAC_CNTRL1_RXBUFFER_SIZE (1U)

1.1.32 Macro: ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_SIZE

Table 32 ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_SIZE

Name	ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_SIZE	
Description	Total size of RAM allocated for transmit buffers for the controller with <Controller Index>..	
Verification method	The macro is generated as total transmit buffer size with 8 byte alignment (value of the macro is a product of the values provided in EthTxBufTotal and EthCtrlTxBufLenBytevalue configuration parameters). <i>Note:</i> If EthCtrlTxBufLenByte is not 8-byte aligned, then it is changed to next 8-byte aligned value. This is done to align the Tx buffers on 8-byte aligned addresses for maximum hardware performance.	
Example(s)	Action	Generated output
	Set EthTxBufTotal to 255 and EthCtrlTxBufLenBytevalue to 1522 for CtrlIdx 0. <i>Note:</i> 8 byte alignment for EthCtrlTxBufLenBytevalue	#define ETH_17_GETHMAC_CNTRL0_TXBUFFER_SIZE (389640U)

<pre> : if (EthCtrlTxBufLenBytevalue mod 8 != 0) then ((1522/8)+1)*8 = 191 * 8 = 1528 </pre>	
<p>Set EthTxBufTotal to 1 and EthCtrlTxBufLenBytevalue to 1522 for CtrlIdx 1</p> <p>Note: 8 byte alignment for EthCtrlTxBufLenBytevalue</p> <pre> : if (EthCtrlTxBufLenBytevalue mod 8 != 0) then ((1522/8)+1)*8 = 191 * 8 = 1528 </pre>	<pre> #define ETH_17_GETHMAC_CNTRL1_TXBUFFER_SIZE (1528U) </pre>
<p>Set EthTxBufTotal to 4 and EthCtrlTxBufLenBytevalue to 0 for CtrlIdx 1</p> <p>Note: 8 byte alignment for EthCtrlTxBufLenBytevalue</p> <pre> : if (EthCtrlTxBufLenBytevalue mod 8 != 0) then ((1522/8)+1)*8 = 191 * 8 = 1528 </pre>	<pre> #define ETH_17_GETHMAC_CNTRL1_TXBUFFER_SIZE (1U) </pre>

1.1.33 Macro: ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_COUNT

Table 33 ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_COUNT

Name	ETH_17_GETHMAC_CNTRL<Controller Index> _TXBUFFER_COUNT	
Description	Selects the total number of transmit buffers for Eth driver for the controller with <Controller Index>.	
Verification method	The macro is generated as a numeric value set in the configuration parameter EthTxBufTotal.	
Example(s)	Action	Generated output
	Set EthTxBufTotal= 255 for CtrlIdx 0.	<pre> #define ETH_17_GETHMAC_CNTRL0_TXBUFFER_COUNT (255U) </pre>
	Set EthTxBufTotal= 0 for CtrlIdx 1.	<pre> #define ETH_17_GETHMAC_CNTRL1_TXBUFFER_COUNT (1U) </pre>

1.1.34 Macro: ETH_17_GETHMAC_CNTRL<Controller Index> _CORE<Core Id>

Table 34 ETH_17_GETHMAC_CNTRL<Controller Index>_CORE<Core Id>

Name	ETH_17_GETHMAC_CNTRL<Controller Index>_CORE<Core Id>	
Description	Controller<Controller Index> configured to <Core Id>.	
Verification method	The macro is generated as STD_ON if available controller is configured to any core else the macro will not be generated.	
Example(s)	Action	Generated output
	Configure controller 0 to Core 1	#define ETH_17_GETHMAC_CNTRL0_CORE1 (STD_ON)
	Configure controller 1 to Core 1	#define ETH_17_GETHMAC_CNTRL1_CORE1 (STD_ON)

1.1.35 Macro: ETH_17_GETHMAC_CNTRL<Controller Index>_CONFIGURED**Table 35 ETH_17_GETHMAC_CNTRL<Controller Index>_CONFIGURED**

Name	ETH_17_GETHMAC_CNTRL<Controller Index>_CONFIGURED	
Description	Controllers configured in the project.	
Verification method	The macro is generated as STD_ON if available controller is configured else the macro will not be generated.	
Example(s)	Action	Generated output
	Configure controller 0.	#define ETH_17_GETHMAC_CNTRL0_CONFIGURED (STD_ON)
	Configure controller 1.	#define ETH_17_GETHMAC_CNTRL1_CONFIGURED (STD_ON)

1.1.36 Macro: ETH_17_GETHMAC_MAX_CNTRL_CORE<Core Id>**Table 36 ETH_17_GETHMAC_MAX_CNTRL_CORE<Core Id>**

Name	ETH_17_GETHMAC_MAX_CNTRL_CORE<Core Id>	
Description	Maximum Controllers allocated to <Core Id> Value 255- represents core is not available in current device.	
Verification method	The macro is generated as a numeric value which is max controllers configured in particular core.	
Example(s)	Action	Generated output
	Configure controller 0 and 1 to Core 0.	#define ETH_17_GETHMAC_MAX_CNTRL_CORE0 (2U)
	No controller configured to Core 1.	#define ETH_17_GETHMAC_MAX_CNTRL_CORE1 (0U)

1.1.37 Macro: ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT**Table 37 ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT**

Name	ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT
Description	Wait time in nanoseconds after a kernel reset in RGMII mode

Verification method	The macro is generated based on the SPB frequency in nanoseconds configured.	
Example(s)	Action	Generated output
	User set value of EthSpbPeriodInNanoSeconds 10. ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT = (35 * \$EthSpbPeriodInNanoSeconds (10))	#define ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT (350U)
	User set value of EthSpbPeriodInNanoSeconds 30. ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT = (35 * \$EthSpbPeriodInNanoSeconds (30))	#define ETH_17_GETHMAC_KRNL_RST_RGMII_WAITCNT (1050U)

1.1.38 Macro: ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT

Table 38 ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT

Name	ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT	
Description	Wait time in nanoseconds after a kernel reset in MII/ RGMII mode.	
Verification method	The macro is generated based on the SPB frequency in nanoseconds configured.	
Example(s)	Action	Generated output
	User set \$EthSpbPeriodInNanoSeconds parameter value 10 ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT = (70 * \$EthSpbPeriodInNanoSeconds(10))	#define ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT (700U)
	User set \$EthSpbPeriodInNanoSeconds parameter value 30 ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT = (70 * \$EthSpbPeriodInNanoSeconds(30))	#define ETH_17_GETHMAC_KRNL_RST_MII_WAITCNT (2100U)

1.1.39 Macro: ETH_17_GETHMAC_DMA_RESET_WAITCYCLE

Table 39 ETH_17_GETHMAC_DMA_RESET_WAITCYCLE

Name	ETH_17_GETHMAC_DMA_RESET_WAITCYCLE	
Description	Number of fSPB cycles to wait after a DMA software reset.	
Verification method	The macro is generated as a numeric value set in the configuration parameter EthDmaSwResetWaitCycle	
Example(s)	Action	Generated output
	User set Default value 4 to EthDmaSwResetWaitCycle parameter.	#define ETH_17_GETHMAC_DMA_RESET_WAITCYCLE (4U)
	User set value 10 to EthDmaSwResetWaitCycle parameter.	#define ETH_17_GETHMAC_DMA_RESET_WAITCYCLE (10U)

1.2 File: Eth_17_GEthMac[_<variant>]_PBcfg.c

Eth_17_GEthMac driver

The generated source file contains all post-build configuration parameters. Post-build time configuration mechanism allows configurable functionality of Ethernet driver that is deployed as object code. The file is generated in 'src' folder.

1.2.1 Structure: Eth_17_GEthMac_CoreCntrlConfigCore<Core Id>[_Variant][Max controllers Configured]

Table 40 Eth_17_GEthMac_CoreCntrlConfigCore<Core Id>[_Variant][Max controllers Configured]

Name	Eth_17_GEthMac_CoreCntrlConfigCore<Core Id>[_Variant][Max controllers Configured]	
Type	Eth_17_GEthMac_CoreCntrlConfigType	
Description	Array of structures to store controller configuration data for a Core.	
Verification method	The generated structure is present in Eth_17_GEthMac[_<variant>]_PBcfg.c file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the structure name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure all the parameter available for the Ethernet controller (variant-unaware).	<pre>static const Eth_17_GEthMac_CoreCntrlConfigType Eth_17_GEthMac_CoreCntrlConfigCore0[1]= { { /*Specifies the Tx[0:3]/Rx[4:7] clock delay in RGMII mode for transmit skew timing*/ (uint32)0, /* Element to store GETH_GPCTL register value for current controller */ (uint32)13, (uint16)1522U, /*Configured Receive Buffer Length*/ (uint16)1528U, /*Receive Buffer Length 8 byte aligned*/ (uint16)1522U, /*Configured Transmit Buffer Length*/ (uint16)1528U, /*Transmit Buffer Length 8 byte aligned*/ /* Properties of Ethernet Controller Bit[0] - Port Select(PS) 0 for 1000Mbps 1 for 10 or 100 Mbps Bit[1] - Speed(FES)</pre>

	<p>0 for 10 Mbps when PS bit is 1 and 1 Gbps when PS bit is 0</p> <p>1 for 100 Mbps when PS bit is 1</p> <p>Bit[2:4] - PhyInterface (000-MII, 100-RMII,001-RGMII)</p> <p>Bit[5] - Mode of the Controller [0 - HALFDUPLEX, 1-FULLDUPLEX]</p> <p>Bit[6] - Tx Interrupt Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>Bit[7] - Rx Interrupt Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>Bit[8] - CRC Stripping Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>*/</p> <p>(uint16)51,</p> <p>(uint8)4U, /*Total Receive Buffer*/</p> <p>(uint8)4U, /*Total Transmit Buffer*/</p> <p>/* MAC address of the controller in network byte order */</p> <p>{</p> <p>(uint8)0x00U,</p> <p>(uint8)0x03U,</p> <p>(uint8)0x19U,</p> <p>(uint8)0x00U,</p> <p>(uint8)0x00U,</p> <p>(uint8)0x02U</p> <p>},</p> <p>/* Eth Controller Index */</p> <p>(uint8)0,</p> <p>/*DEM Id for Ethernet controller hardware test failure*/</p> <p>ETH_17_GETHMAC_DISABLE_DEM_REPORT,</p> <p>/*DEM Id for Ethernet controller Frames Lost Error*/</p> <p>ETH_17_GETHMAC_DISABLE_DEM_REPORT,</p> <p>/*DEM Id for Ethernet controller Frames Alignment Error*/</p> <p>ETH_17_GETHMAC_DISABLE_DEM_REPORT,</p> <p>/*DEM Id for Ethernet controller Frames CRC Error*/</p>
--	---

	<pre> ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Undersize frame Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Oversize frame Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Single collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Multiple collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Late collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, }, }; </pre>
Configure all the parameter available for the Ethernet controller (variant-aware, variant name is 'Petrol').	<pre> static const Eth_17_GEthMac_CoreCntrlConfigType Eth_17_GEthMac_CoreCntrlConfigCore0_ Petrol[1]= { { /*Specifies the Tx[0:3]/Rx[4:7] clock delay in RGMII mode for transmit skew timing*/ (uint32)0, /* Element to store GETH_GPCTL register value for current controller */ (uint32)13, (uint16)1522U, /*Configured Receive Buffer Length*/ (uint16)1528U, /*Receive Buffer Length 8 byte aligned*/ (uint16)1522U, /*Configured Transmit Buffer Length*/ (uint16)1528U, /*Transmit Buffer Length 8 byte aligned*/ /* Properties of Ethernet Controller Bit[0] - Port Select(PS) </pre>

	<p>0 for 1000Mbps</p> <p>1 for 10 or 100 Mbps</p> <p>Bit[1] - Speed(FES)</p> <p>0 for 10 Mbps when PS bit is 1 and 1 Gbps when PS bit is 0</p> <p>1 for 100 Mbps when PS bit is 1</p> <p>Bit[2:4] - PhyInterface (000-MII, 100-RMII,001-RGMII)</p> <p>Bit[5] - Mode of the Controller [0 - HALFDUPLEX, 1- FULLDUPLEX]</p> <p>Bit[6] - Tx Interrupt Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>Bit[7] - Rx Interrupt Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>Bit[8] - CRC Stripping Enable/Disable [0 - Disabled, 1- Enabled]</p> <p>*/</p> <p>(uint16)51,</p> <p>(uint8)4U, /*Total Receive Buffer*/</p> <p>(uint8)4U, /*Total Transmit Buffer*/</p> <p>/* MAC address of the controller in network byte order */</p> <p>{</p> <p>(uint8)0x00U,</p> <p>(uint8)0x03U,</p> <p>(uint8)0x19U,</p> <p>(uint8)0x00U,</p> <p>(uint8)0x00U,</p> <p>(uint8)0x02U</p> <p>},</p> <p>/* Eth Controller Index */</p> <p>(uint8)0,</p> <p>/*DEM Id for Ethernet controller hardware test failure*/</p> <p>ETH_17_GETHMAC_DISABLE_DEM_REPORT,</p> <p>/*DEM Id for Ethernet controller Frames Lost Error*/</p> <p>ETH_17_GETHMAC_DISABLE_DEM_REPORT,</p>
--	---

	<pre> /*DEM Id for Ethernet controller Frames Alignment Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Frames CRC Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Undersize frame Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Oversize frame Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Single collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Multiple collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, /*DEM Id for Ethernet controller Late collision Error*/ ETH_17_GETHMAC_DISABLE_DEM_REPORT, }, }; </pre>
--	--

1.2.1.1 Member: EthSkewDelay

Table 41 EthSkewDelay

Name	EthSkewDelay	
Type	uint32	
Description	Tx/Rx clock delay in RGMII mode for transmit skew timing	
Verification method	<p>The structure member is generated form the configuration parameter EthSkewTxClockDelay and EthSkewRxClockDelay.</p> <p><i>Note: The EthSkewTxClockDelay and EthSkewRxClockDelay skew delay is allowed to be configured only in RGMII mode.</i></p>	
Example(s)	Action	Generated output
	Set EthSkewTxClockDelay to 8 And EthSkewRxClockDelay to 10	(uint32)168
	Set EthSkewTxClockDelay to 1	(uint32)1

And EthSkewRxClockDelay to 0

1.2.1.2 Member: EthGptclRegVal

Table 42 EthGptclRegVal

Name	EthGptclRegVal	
Type	uint32	
Description	Element to store general purpose control register value for the controller.	
Verification method	<p>The structure member is generated from the values configured in EthMdioAlternateInput and other configuration parameters where port selection is done.</p> <p><i>Note: The configuration parameters will be available for selection based on the selected mode (MII/RMII and RGMI).</i></p>	
Example(s)	Action	Generated output
	Set EthPhyInterface = RGMII EthMdioAlternateInput = ALT3_SELECT_P21_3 EthRxcIkInput = ALT0_SELECT_P11_12 EthReceiveData0Input = ALT3_SELECT_NONE EthReceiveData1Input = ALT3_SELECT_NONE EthReceiveData2Input = ALT3_SELECT_NONE EthReceiveData3Input = ALT3_SELECT_NONE	(uint32)3
	Set EthPhyInterface = RMII EthMdioAlternateInput = ALT3_SELECT_P21_3 EthRefClkRMIIInput = ALT3_SELECT_NONE EthCRSDVRMIIInput = ALT3_SELECT_NONE EthReceiveData0Input = ALT0_SELECT_P11_10 EthReceiveData1Input = ALT0_SELECT_P11_9	(uint32)15

1.2.1.3 Member: EthCtrlRxBufLenByte

Table 43 EthCtrlRxBufLenByte

Name	EthCtrlRxBufLenByte
-------------	---------------------

Type	uint16	
Description	Maximum configured receive buffer length (frame length) in bytes	
Verification method	Receive buffer length configured in the configuration parameter EthCtrlRxBufLenByte is generated.	
Example(s)	Action	Generated output
	Set EthCtrlRxBufLenByte to 1522	(uint16)1522
	Set EthCtrlRxBufLenByte to 1	(uint16)1

1.2.1.4 Member: EthCtrlRxBufLenByteAligned

Table 44 EthCtrlRxBufLenByteAligned

Name	EthCtrlRxBufLenByteAligned	
Type	uint16	
Description	Maximum receive buffer length (frame length) aligned to 8 bytes.	
Verification method	Receive buffer length configured in the configuration parameter EthCtrlRxBufLenByteAligned is generated.	
	<p><i>Note:</i> If EthCtrlRxBufLenByte is not 8-byte aligned, then it is changed to next 8-byte aligned value. This is done to align the Rx buffers on 8-byte aligned addresses for maximum hardware performance.</p>	
Example(s)	Action	Generated output
	Set EthCtrlRxBufLenByteAligned to 1522	(uint16)1528
	<p><i>Note:</i> 8 byte alignment for EthCtrlRxBufLenBytevalue: if (EthCtrlRxBufLenBytevalue mod 8 != 0) then ((1522/8)+1)*8 = 1528</p>	
	Set EthCtrlRxBufLenByteAligned to 231	(uint16)232
	<p><i>Note:</i> 8 byte alignment for EthCtrlRxBufLenBytevalue: if (EthCtrlRxBufLenBytevalue mod 8 != 0) then ((231/8)+1)*8 = 232</p>	

1.2.1.5 Member: EthCtrlTxBufLenByte

Table 45 EthCtrlTxBufLenByte

Name	EthCtrlTxBufLenByte	
Type	uint16	
Description	Maximum configured transmit buffer length (frame length) in bytes	
Verification method	Transmit buffer length configured in the configuration parameter EthCtrlTxBufLenByte is generated.	
Example(s)	Action	Generated output
	Set EthCtrlTxBufLenByte to 1522	(uint16)1522
	Set EthCtrlTxBufLenByte to 1	(uint16)1

1.2.1.6 Member: EthCtrlTxBufLenByteAligned

Table 46 EthCtrlTxBufLenByte

Name	EthCtrlTxBufLenByteAligned	
Type	uint16	
Description	Maximum transmit buffer length (frame length) aligned to 8 bytes.	
Verification method	Receive buffer length configured in the configuration parameter EthCtrlTxBufLenByteAligned is generated. <i>Note: If EthCtrlTxBufLenByte is not 8-byte aligned, then it is changed to next 8-byte aligned value. This is done to align the Rx buffers on 8-byte aligned addresses for maximum hardware performance.</i>	
Example(s)	Action	Generated output
	EthCtrlTxBufLenBytevalue to 1522 <i>Note: 8 byte alignment for EthCtrlTxBufLenBytevalue: if $(EthCtrlTxBufLenBytevalue \bmod 8 \neq 0)$ then $((1522/8)+1)*8 = 1528$</i>	(uint16)1528
	Set EthCtrlTxBufLenByte to 231 <i>Note: 8 byte alignment for EthCtrlTxBufLenBytevalue: if $(EthCtrlTxBufLenBytevalue \bmod 8 \neq 0)$ then $((231/8)+1)*8 = 232$</i>	(uint16)232

1.2.1.7 Member: EthCntrlProperties

Table 47 EthCntrlProperties

Name	EthCntrlProperties	
Type	uint16	
Description	Properties of Ethernet Controller.	
Verification method	Ethernet controller properties is generated from the values configured in EthSpeed, EthPhyInterface, EthOpMode, EthCtrlEnableTxInterrupt, EthCtrlEnableRxInterrupt and EthCtrlEnableCrcStripping.	
Example(s)	Action	Generated output
	Set EthSpeed = ETH_100MBPS EthPhyInterface = RMII EthOpMode = FULLDUPLEX EthCtrlEnableRxInterrupt = True EthCtrlEnableTxInterrupt = True EthCtrlEnableCrcStripping = True	(uint16)499
	Set EthSpeed = ETH_1000MBPS EthPhyInterface = RGMII EthOpMode = FULLDUPLEX EthCtrlEnableRxInterrupt = False EthCtrlEnableTxInterrupt = False EthCtrlEnableCrcStripping = False	(uint16)36,

1.2.1.8 Member: EthDemAccess

Table 48 EthDemAccess

Name	EthDemAccess	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_ACCESS Failure	
Verification method	DEM Id is generated for ETH_E_ACCESS as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note: <container Id> is the DemEventParameter container in DEM module.</i>	
Example(s)	Action	Generated output

Eth_17_GEthMac driver

EthDemEventParameterRefs container and ETH_E_ACCESS is configured.	DemConf_DemEventParameter_ETH_E_ACCESS
EthDemEventParameterRefs container and ETH_E_ACCESS is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.9 Member: EthDemFramesLost

Table 49 EthDemFramesLost

Name	EthDemFramesLost	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_RX_FRAMES_LOST Failure.	
Verification method	DEM Id is generated for ETH_E_RX_FRAMES_LOST as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note: <container Id> is the DemEventParameter container in DEM module.</i>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_RX_FRAMES_LOST is configured.	DemConf_DemEventParameter_ETH_E_RX_FRAMESLOST
	EthDemEventParameterRefs container and ETH_E_RX_FRAMES_LOST is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.10 Member: EthDemAlignment

Table 50 EthDemAlignment

Name	EthDemAlignment	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_ALIGNMENT Failure	
Verification method	DEM Id is generated for ETH_E_ALIGNMENT as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note: <container Id> is the DemEventParameter container in DEM module.</i>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and	DemConf_DemEventParameter_ETH_E_ALIGNMENT

Eth_17_GEthMac driver

ETH_E_ALIGNMENT is configured.	
EthDemEventParameterRefs container and ETH_E_ALIGNMENT is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.11 Member: EthDemCRC

Table 51 EthDemCRC

Name	EthDemCRC	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_CRC Failure	
Verification method	DEM Id is generated for ETH_E_CRC as DemConf_DemEventParameter_ <container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note: <container Id> is the DemEventParameter container in DEM module.</i>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_CRC is configured.	DemConf_DemEventParameter_ETH_E_CRC
	EthDemEventParameterRefs container and ETH_E_CRC is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.12 Member: EthDemUndersize

Table 52 EthDemUndersize

Name	EthDemUndersize	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_UNDERSIZEFRAME Failure	
Verification method	DEM Id is generated for ETH_E_UNDERSIZEFRAME as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note: <container Id> is the DemEventParameter container in DEM module.</i>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_UNDERSIZEFRAME is configured.	DemConf_DemEventParameter_ETH_E_UNDERSIZEFRAME
		ETH_17_GETHMAC_DISABLE_DEM_REPORT

EthDemEventParameterRefs container and ETH_E_UNDERSIZEFRAME is not configured.	
--	--

1.2.1.13 Member: EthDemOversize

Table 53 EthDemOversize

Name	EthDemOversize	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_OVERSIZEFRAME Failure	
Verification method	<p>DEM Id is generated for ETH_E_OVERSIZEFRAME as DemConf_DemEventParameter _<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT.</p> <p><i>Note: <container Id> is the DemEventParameter contained in DEM module.</i></p>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_OVERSIZEFRAME is configured.	DemConf_DemEventParameter_ETH_E_OVERSIZEFRAME
	EthDemEventParameterRefs container and ETH_E_OVERSIZEFRAME is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.14 Member: EthDemSingleCollision

Table 54 EthDemSingleCollision

Name	EthDemSingleCollision	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_SINGLE_COLLISION Failure	
Verification method	<p>DEM Id is generated for ETH_E_SINGLE_COLLISION as DemConf_DemEventParameter _<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT.</p> <p><i>Note: <container Id> is the DemEventParameter contained in DEM module.</i></p>	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_SINGLE_COLLISION is configured.	DemConf_DemEventParameter_ETH_E_SINGLECOLLISION
		ETH_17_GETHMAC_DISABLE_DEM_REPORT

EthDemEventParameterRefs container and ETH_E_SINGLE_COLLISION is not configured.
--

1.2.1.15 Member: EthDemMultipleCollision

Table 55 EthDemMultipleCollision

Name	EthDemMultipleCollision	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_MULTIPLE_COLLISION Failure	
Verification method	DEM Id is generated for ETH_E_MULTIPLE_COLLISION as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note:</i> <container Id> is the DemEventParameter contained in DEM module.	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_MULTIPLE_COLLISION is configured.	DemConf_DemEventParameter_ETH_E_MULTIPLE COLLISION
	EthDemEventParameterRefs container and ETH_E_MULTIPLE_COLLISION is not configured.	ETH_17_GETHMAC_DISABLE_DEM_REPORT

1.2.1.16 Member: EthDemLateCollision

Table 56 EthDemLateCollision

Name	EthDemLateCollision	
Type	Dem_EventIdType	
Description	DEM Id for ETH_E_LATE_COLLISION Failure	
Verification method	DEM Id is generated for ETH_E_LATE_COLLISION as DemConf_DemEventParameter_<container Id>. If DEM is not configured, then DEM Id is generated as ETH_17_DISABLE_DEM_REPORT. <i>Note:</i> <container Id> is the DemEventParameter contained in DEM module.	
Example(s)	Action	Generated output
	EthDemEventParameterRefs container and ETH_E_LATE_COLLISION is configured.	DemConf_DemEventParameter_ETH_E_LATECOLLISION
		ETH_17_GETHMAC_DISABLE_DEM_REPORT

EthDemEventParameterRefs container and ETH_E_LATE_COLLISION is not configured.
--

1.2.1.17 Member: EthRxBufTotal

Table 57 EthRxBufTotal

Name	EthRxBufTotal	
Type	uint8	
Description	Total number of receive buffers	
Verification method	Total number of receive buffers configured in the configuration parameter EthRxBufTotal is generated.	
Example(s)	Action	Generated output
	Set EthRxBufTotal to 100	(uint8)100
	Set EthRxBufTotal 255	(uint8)255

1.2.1.18 Member: EthTxBufTotal

Table 58 EthTxBufTotal

Name	EthTxBufTotal	
Type	uint8	
Description	Total number of transmit buffers	
Verification method	Total number of transmit buffers configured in the configuration parameter EthTxBufTotal is generated.	
Example(s)	Action	Generated output
	Set EthTxBufTotal to 100	(uint8)100
	Set EthTxBufTotal 255	(uint8)255

1.2.1.19 Member: EthCntrlIdx

Table 59 EthCntrlIdx

Name	EthCntrlIdx	
Type	uint8	
Description	Controller index of Ethernet.	
Verification method	The structure member is generated for value configured in EthCntrlIdx	
Example(s)	Action	Generated output
	Set EthCntrlIdx = 0	(uint8)0
	Set EthCntrlIdx = 1	(uint8)1

1.2.1.20 Member: EthMacAddress [6]

Table 60 EthMacAddress [6]

Name	EthMacAddress [6]	
Type	Uint8	
Description	PHY MAC address in Network Byte order	
Verification method	The generated structure member contains an array entry for user-configured PHY MAC address in Network Byte order as configured in the configuration parameter EthCtrlPhyAddress.	
Example(s)	Action	Generated output
	Set EthCtrlPhyAddress to 00:03:19:00:00:01	<pre>{ (uint8)0x00U, (uint8)0x03U, (uint8)0x19U, (uint8)0x00U, (uint8)0x00U, (uint8)0x01U },</pre>
	Set EthCtrlPhyAddress to 11:11:11:11:11:11	<pre>{ (uint8)0x11U, (uint8)0x11U, (uint8)0x11U, (uint8)0x11U, (uint8)0x11U, (uint8)0x11U },</pre>

1.2.2 Structure: Eth_17_GEthMac_ConfigCore<Core Id>[_Variant]

Table 61 Eth_17_GEthMac_ConfigCore<Core Id>[_Variant]

Name	Eth_17_GEthMac_ConfigCore<Core Id>[_Variant]	
Type	Eth_17_GEthMac_CoreConfigType	
Description	Structure to store core specific configuration data.	
Verification method	The generated structure is present in Eth_17_GEthMac[_<variant>]_PbCfg.c file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the structure name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure controller 0 and controller 1 to core 1.	<pre>static const Eth_17_GEthMac_CoreConfigType Eth_17_GEthMac_ConfigCore1 =</pre>

	<pre>{ (Eth_17_GEthMac_CoreCntrlConfigType*) Eth_17_GEthMac_CoreCntrlConfigCore1, 2U /* Maximum controllers allocated to core1 */ };</pre>
Configure controller 0 to core 0.(variant-aware, variant name is 'Petrol').	<pre>static const Eth_17_GEthMac_CoreConfigType Eth_17_GEthMac_ConfigCore0_Petrol = { (Eth_17_GEthMac_CoreCntrlConfigType*) Eth_17_GEthMac_CoreCntrlConfigCore0_ Petrol, 1U /* Maximum controllers allocated to core0 */ };</pre>

1.2.2.1 Member: EthCoreCntrlPtr

Table 62 EthCoreCntrlPtr

Name	EthCoreCntrlPtr	
Type	Eth_17_GEthMac_CoreCntrlConfigType	
Description	Pointer to the configuration of controller allocated to that core.	
Verification method	The structure member is generated with address which stores controller configuration.	
Example(s)	Action	Generated output
	Configure controller 0 to Core 1	(Eth_17_GEthMac_CoreCntrlConfigType*) Eth_17_GEthMac_CoreCntrlConfigCore1,
	Configure controller 0 to core 0.(variant-aware, variant name is 'Petrol').	(Eth_17_GEthMac_CoreCntrlConfigType*) Eth_17_GEthMac_CoreCntrlConfigCore0_ Petrol,

1.2.2.2 Member: EthMaxControllers

Table 63 EthMaxControllers

Name	EthMaxControllers	
Type	uint8	
Description	Maximum controllers allocated to the core.	
Verification method	The structure member is generated as max controllers configured to the core.	
Example(s)	Action	Generated output
	Configure controller 0 and controller 1 to Core 1	2U

Configure controller 0 to core 0.	1U
-----------------------------------	----

1.2.3 Structure: Eth_17_GEthMac_Config[_Variant]

Table 64 Eth_17_GEthMac_Config[_Variant]

Name	Eth_17_GEthMac_Config[_Variant]	
Type	Eth_17_GEthMac_ConfigType	
Description	Ethernet driver configuration root structure.	
Verification method	The generated structure is present in Eth_17_GEthMac[_<variant>]_PBCfg.c file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the structure name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure controller 0 to Core0 and controller 1 to core 1.	<pre>const Eth_17_GEthMac_ConfigType Eth_17_GEthMac_Config = { /* starting address of Core<x> Configuration data */ { (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore0, (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore1, NULL_PTR }, /* Address of index mapping array */ (uint8*) Eth_17_GEthMac_ControllerIndexMap };</pre>
	Configure controller 0 and controller 1 to core 0.(variant-aware, variant name is 'Petrol').	<pre>const Eth_17_GEthMac_ConfigType Eth_17_GEthMac_Config_Petrol = { /* starting address of Core<x> Configuration data */ { (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore0_ Petrol, NULL_PTR,</pre>

	<pre> NULL_PTR }, /* Address of index mapping array */ (uint8*) Eth_17_GEthMac_ControllerIndexMap _Petrol }; </pre>
--	---

1.2.3.1 Member: EthCoreAdd[ETH_17_GETHMAC_MAX_CORES]

Table 65 EthMaxControllers

Name	EthCoreAdd[ETH_17_GETHMAC_MAX_CORES]	
Type	Eth_17_GEthMac_CoreConfigType *	
Description	Array to store starting address of core configuration data.	
Verification method	The generated structure member is present in the Eth_17_GEthMac_Config[_<variant>] structure. If a controller <y> is allocated at least one Core<x>, then the element shall be generated as '&Eth_17_GEthMac_ConfigCore<x>' else 'NULL_PTR' is generated.(x in range 0 to 5) and (y in range 0 or 1) depending on the derivative.	
Example(s)	Action	Generated output
	Configure controller 0 to Core0 and controller 1 to core 1.	<pre> /* starting address of Core<x> Configuration data */ { (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore0, (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore1, NULL_PTR }, </pre>
	Configure controller 0 and controller 1 to core 0.(variant-aware, variant name is 'Petrol').	<pre> /* starting address of Core<x> Configuration data */ { (Eth_17_GEthMac_CoreConfigType*) &Eth_17_GEthMac_ConfigCore0_ Petrol, NULL_PTR, NULL_PTR }, </pre>

1.2.3.2 Member: EthNodeIdxmapPtr

Table 66 EthNodeIdxmapPtr

Name	EthNodeIdxmapPtr	
Type	uint8	
Description	Pointer to the array index of the controller in the current core.	
Verification method	The structure member is generated as pointer to the array index of the controller.	
Example(s)	Action	Generated output
	Configure controller 0 and controller 1 to Core 0.	(uint8*)Eth_17_GEthMac_Controller IndexMap
Example(s)	Configure controller 0 and controller 1 to core 0.(variant-aware, variant name is 'Petrol')..	(uint8*) Eth_17_GEthMac_ControllerIndexMap _Petrol

1.2.4 Array: Eth_17_GEthMac_ControllerIndexMap[_Variant][Max controllers Configured]

Table 67 Eth_17_GEthMac_ControllerIndexMap[_Variant][Max controllers Configured]

Name	Eth_17_GEthMac_ControllerIndexMap[_Variant][Max controllers Configured]	
Type	uint8	
Description	Array to store index of the controller in the allocated core.	
Verification method	Eth_17_GEthMac_ControllerIndexMap [<x>] = Index of (Controller = <x>) in the allocated core.<Variant> indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure controller 0 to Core0 and controller 1 to core 1.	static const uint8 Eth_17_GEthMac_ControllerIndexMap [2] = { 0x0U, 0x0U, };
Example(s)	Configure controller 0 and controller 1 to core 0.(variant-aware, variant name is 'Petrol').	static const uint8 Eth_17_GEthMac_ControllerIndexMap_ Petrol [2] = { 0x0U, 0x1U,

};

1.3 File: Eth_17_GEthmac[_<variant>]_PBcfg.h

The generated header file contains the declaration of the root configuration structure. Post-build time configuration mechanism allows configurable functionality of Ethernet driver that is deployed as object code. The file is generated in 'inc' folder.

1.3.1 Structure: Eth_17_GEthMac_Config[_<variant>]

Table 68 Eth_17_GEthMac_Config[_<varaint>]

Name	Eth_17_GEthMac_Config[_<variant>]	
Type	Eth_17_GEthMac_ConfigType	
Description	Declaration of root configuration structure of Ethernet driver which will be used during initialization.	
Verification method	The generated structure is present in Eth_17_GEthMac[_<variant>]_PBcfg.h file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the structure name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure all the parameter available for the Ethernet controller (variant-unaware).	extern const Eth_17_GEthMac_ConfigType Eth_17_GEthMac_Config;
	Configure all the parameter available for the Ethernet controller (variant-aware, variant name is 'Petrol').	extern const Eth_17_GEthMac_ConfigType Eth_17_GEthMac_Config_Petrol;

Revision history

Major changes since the last revision

Date	Version	Description
02-12-2020	V4.0	Released version with review comments incorporated
27-11-2020	V3.1	Updated configuration macros ETH_17_GETHMAC_KRNLRST_RGMII_WAITCNT, ETH_17_GETHMAC_KRNLRST_MII_WAITCNT and ETH_17_GETHMAC_DMA_RESET_WAITCYCLE
18-07-2019	v3.0	Released version with review comments incorporated
11-07-2019	v2.1	Updated configuration macros and configuration structure for multicore changes.
28-02-2019	v1.10.0_2.0	Added PBCfg.h file.
25-02-2019	v1.10.0_1.0	Released Version.
21-02-2019	v1.10.0_0.1	Initial Version

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2020-12-02

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2020 Infineon Technologies AG.

All Rights Reserved.

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

Doc_Number

IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.