OVERVIEW

The single-chip PCI Express based 88E8053 device integrates the Marvell® market-leading Gigabit PHY with the proven Marvell Gigabit MAC and SERDES cores, delivering an ultra-small form factor and high performance. Delivered with the industry's most comprehensive software driver suite, this Yukon device is ideally suited for LAN on motherboard (LOM) and Network Interface Card (NIC) applications. The 88E8053 device is compliant with the PCI Express 1.0a specification. Offered in a 9 x 9 mm, 64-pin QFN package, the 88E8053 reduces board space required for Gigabit LOM implementation significantly.

The device is optimized for maximum throughput and low CPU utilization. A 48 KB on-chip buffer eliminates the need for any external memory. Packet processing tasks such as TCP segmentation, VLAN insertion and removal, TCP/UDP/IP checksum calculation and checking are all performed on-chip. These offloads along with interrupt moderation schemes reduce CPU utilization and improve the overall system performance.

The 88E8053 device incorporates advanced power management schemes, enabling energy efficient operation. With features such as Wake on LAN and Smart Power Down in the absence of link it is well suited for client applications including mobile PCs.

The 88E8053 Yukon device incorporates the Marvell Virtual Cable Tester™ (VCT™) technology for advanced cable diagnostics. VCT enables IT managers to pinpoint the location of cabling issues down to a meter or less, reducing network installation and support costs.

The device comes with a comprehensive suite of software device drivers for all desktop operating systems, including Microsoft® Windows® 98/Me, NT, 2000, and XP, Linux, and Novell Netware. A complete hardware reference design is provided for a quick implementation.

FEATURES

PCI EXPRESS FEATURES

- PCI Express base specification 1.0a compliant
- x1 PCI Express interface with 2.5 GHz signaling
- Active state power management (L0s) support
- Advanced error reporting

MAC / PHY FEATURES

- Configurable 48 KB deep buffer
- Descriptor bursting and caching
- Message signaled interrupts
- TCP segmentation offload / Large-send support
- On-chip VLAN insertion and removal
- TCP, IP, UDP Checksum offload
- Interrupt moderation
- Jumbo frame support
- Compliant to 802.3x flow control support
- IEEE 802.1p and 802.1q support
- 10/100/1000 IEEE 802.3 compliant
- Automatic MDI/MDIX crossover at all speeds

MANAGEABILITY

- Wake On LAN (WOL) power management support
- Compliant to ACPI 2.0 specification
- Out of the box WOL support
- Wake On Link
- Serial Peripheral Interface (SPI) for remote boot (PXE 2.1)
- Smart power down when link is not detected
- Marvell Virtual Cable Tester™ (VCT) for advanced cable diagnostics

OTHER FEATURES

- LOM disable pin
- Power regulator outputs for 2.5V and 1.2V supplies
- Two Wire Serial Interface (TWSI) for VPD EEPROM
- 9 mm x 9 mm, 64-pin QFN package

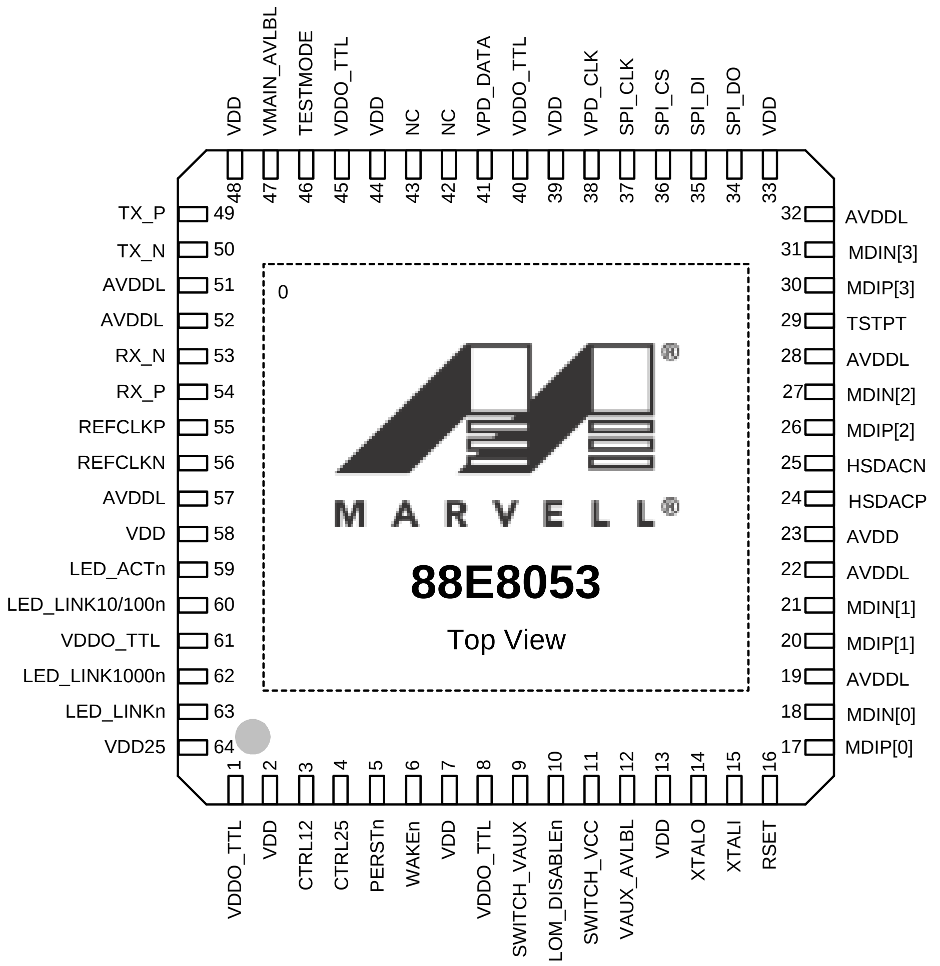
Table of Contents

Section 1. Signal Description

1.1 64-Pin QFN Pinout (Top View)

The 88E8053 device is manufactured in a 64-pin QFN, 9 x 9 mm package.

Figure 1: 88E8053 64-Pin QFN Package (Top View)



1.2 Pin Description

1.2.1 Pin Type Definitions

Pin Type	Definition
H	Input with hysteresis
I/O	Input and output
I	Input only
O	Output only
PU	Internal pull-up
PD	Internal pull-down
D	Open drain output
Z	Tri-state output
mA	DC sink capability

Table 1: PCI Express Signals

88E8053 Pin #	Pin Name	Pin Type	Description
49 50	TX_P TX_N	O Analog	PCI Express Transmit line (positive and negative pair). 2.5 GHz low-voltage pair.
54 53	RX_P RX_N	I Analog	PCI Express Receive line (positive and negative pair). 2.5 GHz low-voltage pair.
6	WAKEn	O	PCI Express wake signal. Driven low to re-activate the PCI Express lin hiearchy's main power rails and reference clocks. (Open collector, active low.) Multiplexed to the same pin as PCI signal PMEn.
55 56	REFCLKP REFCLKN	I Analog	PCI Express platform reference clock (differential pair of positive and negative signal lines). 100 MHz low-voltage interface.
5	PERSTn	I	PCI Express fundamental reset. Asserted 100 ms after power rails are within specifications.

Table 2: Media Dependent Interface (PHY)

88E8053 Pin #	Pin Name	Pin Type	Description
17 18	MDIP[0] MDIN[0]	I/O, D	Media Dependent InterfacePositive/Negative[0]. In 1000BASE-T mode in MDI configuration, MDIP/N[0] correspond to BI_DAP/N. In MDIX configuration, MDIP/N[0] correspond to BI_DBP/N. In 100BASE-TX and 10BASE-T modes in MDI configuration, MDIP/N[0] are used for the transmit pair. In MDIX configuration, MDIP/N[0] are used for the receive pair. MDIP/N[0] should be tied to ground if not used.
20 21	MDIP[1] MDIN[1]	I/O, D	Media Dependent InterfacePositive/Negative[1]. In 1000BASE-T mode in MDI configuration, MDIP/N[1] correspond to BI_DAP/N. In MDIX configuration, MDIP/N[1] correspond to BI_DBP/N. In 100BASE-TX and 10BASE-T modes in MDI configuration, MDIP/N[1] are used for the transmit pair. In MDIX configuration, MDIP/N[1] are used for the receive pair. MDIP/N[1] should be tied to ground if not used.
26 27	MDIP[2] MDIN[2]	I/O, D	Media Dependent InterfacePositive/Negative[2]. In 1000BASE-T mode in MDI configuration, MDIP/N[2] correspond to BI_DAP/N. In MDIX configuration, MDIP/N[2] correspond to BI_DBP/N. In 100BASE-TX and 10BASE-T modes, MDIP/N[2] are not used. MDIP/N[2] should be tied to ground if not used.
30 31	MDIP[3] MDIN[3]	I/O, D	Media Dependent InterfacePositive/Negative[3]. In 1000BASE-T mode in MDI configuration, MDIP/N[3] correspond to BI_DAP/N. In MDIX configuration, MDIP/N[3] correspond to BI_DBP/N. In 100BASE-TX and 10BASE-T modes, MDIP/N[3] are not used. MDIP/N[3] should be tied to ground if not used.

Table 3: TWSI Interface (For Connection to TWSI EEPROM)

88E8053 Pin #	Pin Name	Pin Type	Description
38	VPD_CLK	O, D, PU	TWSI bus clock line to serial EEPROM (with VPD/Boot data). VPD_CLK contains an internal pull-up resistor.
41	VPD_DATA	Bi-Dir, PU	TWSI bus data line to serial EEPROM (with VPD/Boot data). VPD_DATA contains an internal pull-up resistor.

Table 4: SPI Flash Memory Interface

88E8053 Pin #	Pin Name	Pin Type	Description
34	SPI_DO	O	Data line leading to the SPI Flash Memory. SPI_DO contains an internal pull-up resistor.
35	SPI_DI	I	Data line coming from the SPI Flash Memory. SPI_DI contains an internal pull-up resistor.
37	SPI_CLK	O	Clock line for SPI Interface. SPI_CLK contains an internal pull-up resistor.
36	SPI_CS	O	Chip select for SPI Flash Memory. SPI_CS contains an internal pull-up resistor.

Table 5: Main Clock Interface (PHY)

88E8053 Pin #	Pin Name	Pin Type	Description
15	XTALI	I	Input from 25 MHz Crystal or Oscillator.
14	XTALO	O	Output to 25 MHz Crystal.

Table 6: Analog (PHY)

88E8053 Pin #	Pin Name	Pin Type	Description
10	LOM_ DISABLEn	I	Used in LOM applications. The LOM_DISABLEn pin is active low 0 = LAN on motherboard (LOM) disabled 1 = LOM enabled
12	VAUX_AVLBL	I	VAUX available signal.
11	SWITCH_VCC	O	Switch to VCC.
47	VMAIN_AVLBL	I	VMAIN available signal.
9	SWITCH_VAUX	O	Switch to Vaux.
24 25	HSDACP HSDACN	Analog O, D	PHY Test pin. These pins are used for debug only. If debug is not important and there are board space constraints, these pins should be left floating.
16	RSET	Analog I	PHY Constant voltage reference. External 5.0 kΩ 1% resistor connection to VSS.
4	CTRL25	O Analog	Regulator Control. This signal controls an external PNP transistor to generate the 2.5V power supply.
3	CTRL12	O Analog	Regulator Control. This signal controls an external PNP transistor to generate the 1.2V power supply.

Table 7: LED Interface

88E8053 Pin #	Pin Name	Pin Type	Description
59	LED_ACTn	TTL, D	Parallel LED activity indicator. Active low.
60	LED_ LINK10/100n	TTL, D	Parallel LED output for 100BASE-T link or speed. If the LED_LINK10/100 pin is active, it indicates 100 Mbps. Active low.
62	LED_LINK1000n	TTL, D	Parallel LED output for 1000BASE-T link. Active low.
63	LED_LINKn	TTL, D	Parallel LED output for 10/100/1000BASE-T link. Active low.

Table 8: Test Pins

88E8053 Pin #	Pin Name	Pin Type	Description
46	TESTMODE	I, PD	Selection of internal Test. (Default pull-down.)
29	TSTPT	O	Analog test point.

Table 9: Power & Ground

88E8053 Pin #	Pin Name	Pin Type	Description
64	VDD25	Power	Power to TTL I/Os. 2.5V
19 22 28 32 51 52 57	AVDDL	Power	Analog Power. 2.5V - Copper
23	AVDD	Power	Analog Power. 2.5V (For 3GIOs)
1 8 40 45 61	VDDO_TTL	Power	Power to TTL I/Os. 3.3V (Pin 8 can be used as power to VAUX 3.3V - see Application Note for details.)
2 7 13 33 39 44 48 58	VDD	VDD	Power. 1.2V
0	EPAD	Ground	Ground.

Table 10: No Connect

88E8053 Pin #	Pin Name	Pin Type	Description
42 43	NC	--	No connect. These pins must be left floating.

Section 2. Functional Description

2.1 Overview

This Single Link Gigabit Ethernet Controller device comes with a PCI Express interface and Gigabit Ethernet cable connectivity. The device integrates PCI Express interface, BMUs, RAM, MAC, PHY, and SERDES cores.

An optional 128/256 kByte SPI Flash Memory holds Bootcode and Configuration Data. The Configuration Data is read by the SPI Flash Memory Loader after POWER ON RESET. All writable registers on the PCI device, may be reloaded from the SPI Flash Memory. The SPI Flash Memory is required if other ROM options are not available for bootcode (PXE). The VPD data are stored within an onboard TWSI EEPROM. In absence of this TWSI EEPROM the VPD data may be stored within the SPI Flash Memory.

The PCI device is controlled by the system's CPU through the PCI Target Interface. The description of all accessible registers is concentrated in the chapters **Configuration Register File** and **Control Register File**. Most registers are controlled only at initialization time.

Receive/transmit data and descriptors are transferred to/from system memory over the PCI Interface controlled by the Buffer Management Units (BMU). The GE Link has a Receive Queue, and a asynchronous Transmit Queue. These two queues run independently. The BMU manages the data transfer to/from system memory. This is controlled by the system's CPU through descriptors allocating memory buffers. Descriptors are organized in chained lists.

The PCI FIFO provides data/space for burst transfers. The Rambuffer Control Logic organizes programmable Rambuffer areas in the internal SRAM as FIFOs for buffering of receive/transmit data (e.g. in order to prevent receive overflows). It controls the dataflow between its PCI FIFO and MAC FIFO through the allocated Rambuffer. Packets may be transferred in Flow Through Mode or Store & Forward Mode.

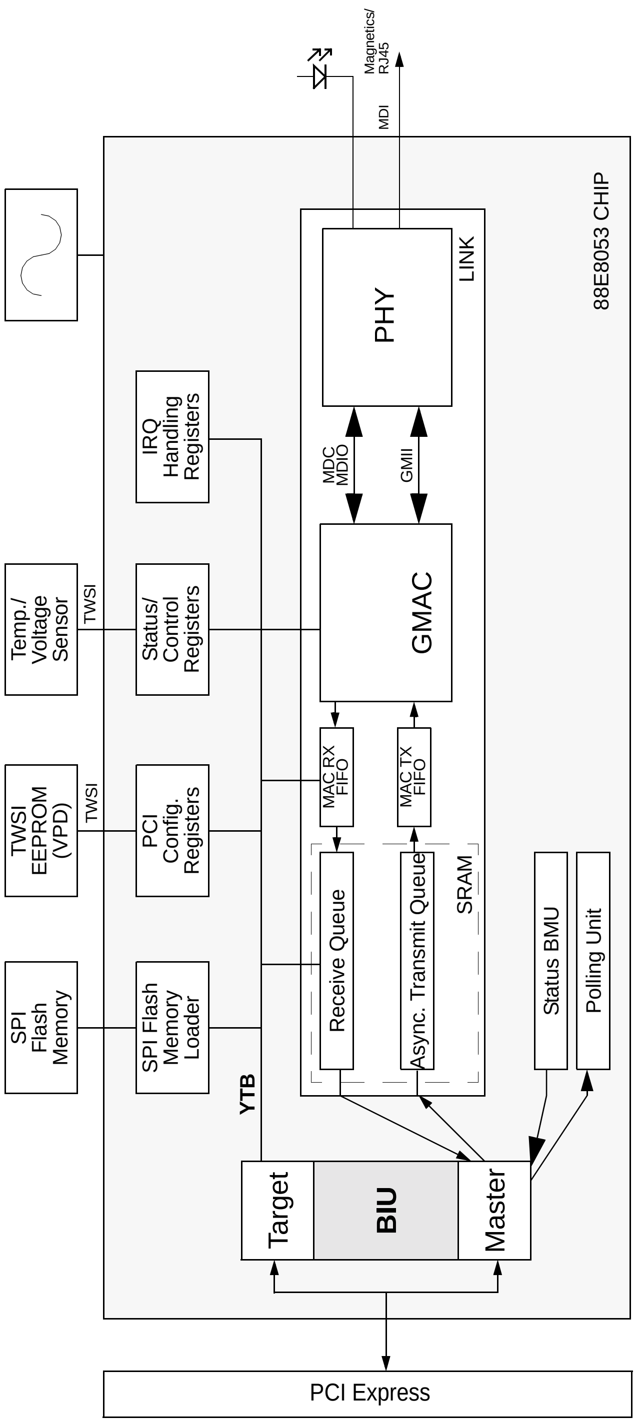
PCI Accesses are arbitrated in a hierarchical priority scheme. Transfer length is optimized for Cache Line Sizes and best PCI Command usage (descriptors and data). The PCI Master supports fully misaligned transfers of Receive/Transmit Data (descriptors must be located at dword boundaries).

The RAM Interface arbitrates RAM accesses requested by the Rambuffer Control Logics in a rotating priority scheme generally. In some cases the priority scheme is adapted dynamically (e.g. risk of receive overflow, Rate Control on transmit). Re-arbitration takes place on chunk level. Chunk length is programmable, default is 32 qwords.

For Transmit Packets a TCP/IP Checksum may be calculated and inserted, on Receive Packets, TCP/IP Checksum is calculated. TCP/IP Checksum Generation/Checking is controlled by the descriptors. TCP/UDP Checksum generation and insertion can be handled also for large fragmented packets. For UDP checksum is calculated over all the appropriate packets and is inserted into the first packet of the sequence.

Figure 2 shows the device block diagram.

Figure 2: Block Diagram



2.2 PCI-Express Features

Table 11 lists the features of the PCI-Express interface.

Table 11: General PCI-Express Features

Feature Name	Description
Standard Compliance	• Compliant with *PCI-Express Base 1.0a Specification*.
PCI-Express port	• 2.5 GHz Signaling. • X1 link width. • Width strapped on reset. • Support link reversal and lane polarity reversal.
Master transaction types	• Support of up to 8 outstanding NP requests as a master (requester). • All Memory transactions, except lock related.
Target transaction types	• Support of 2 outstanding non-posted requests as a target (completer). • All Memory transactions, except lock related. • I/O transactions - supported only when working in legacy endpoint mode. • Configuration transactions - type0 only. • Support up to 8B target accesses.
Message support	• Interrupt messages. • Error messages. • PM related. • Hot-plug related. • Not supported - lock related and vendor specific messages.
Configuration Space.	• Extended 4 KB PCI-Express configuration space. • Single Function Device. • External configuration register file.
Interrupts	• Support of both MSI and interrupt messages. • External BIU master agent is responsible for MSI generation.
Error Reporting	• Full support of PCI-Express base-line error reporting. • Full support of Advanced error reporting capability. • Three error severity levels: Correctable, Uncorrectable - Non-Fatal and Uncorrectable - Fatal. • Header logging and pointer to first uncorrectable error. • Programmable error severity. • PCI error mapping - Mapping of errors to PCI error reporting mechanism.
Address Space	• Three 64bit memory BARs for internal registers access. • One I/O BAR for internal registers access. • Expansion ROM BAR.
Virtual Channels	• Support of baseline TC0-VC0 mapping. • One virtual channel (VC) hardware resource.

Table 11: General PCI-Express Features

Feature Name	Description
Power Management	• Supported SW directed PM states: L0, L1, L2, L3 • Supported Active State Link PM states: L0s-Rx. • Support of wake event generation from all device PM states including $D3_{hot}$. • Wake event signalling by WAKE# signal mechanisms.

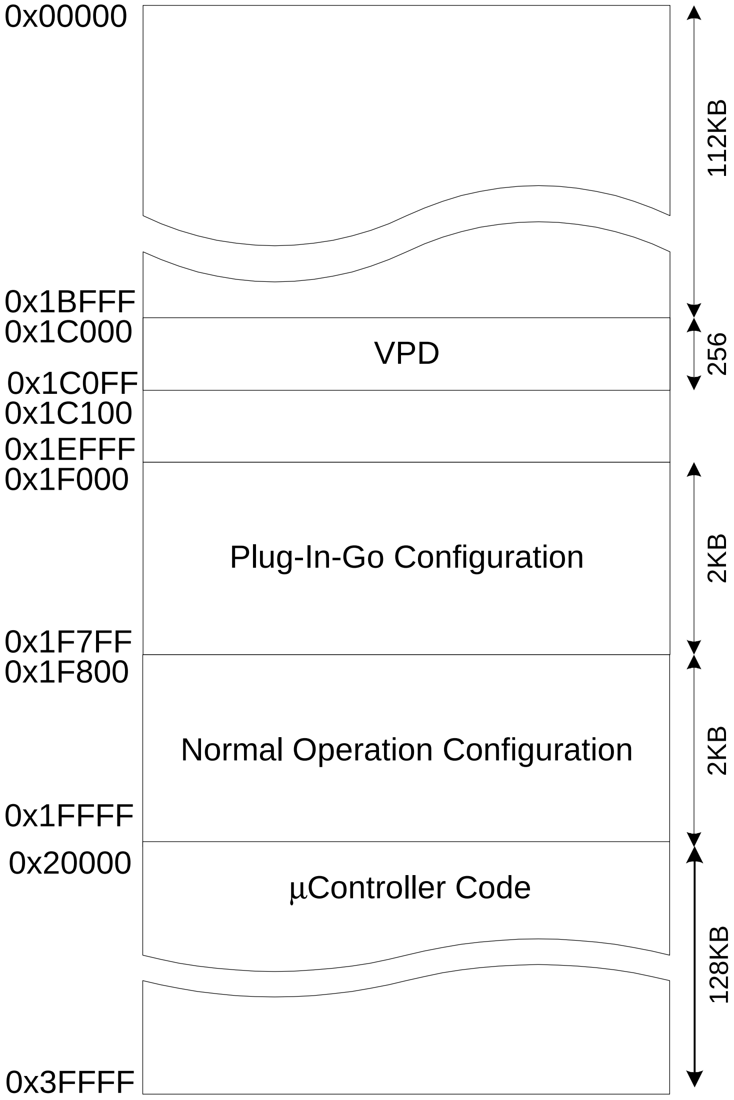
Table 12: Main PCI-Express Parameters.

Feature Name	Description
Max Payload Size	• 128B
Maximum Read Request Size	• Up to 4 KB
RCB - Read Completion Boundary parameter.	• 128B

2.3 SPI Flash Memory

The SPI Flash Memory has a size of 128/256 KByte. For future applications SPI Flash Memories up to 1 MByte can be handled.

Figure 3: Address Space Organization of SPI Flash Memory



The SPI Flash Memory may be mapped into the memory address space with sizes of 16 KB, 32 KB, 64 KB or 128 KB. The page size is defined by **Pagesize<1:0>** (**Our Register 1**).

If **En Eprom** (**Our Register 1**) is not set, the **Expansion Rom Base Address Register** is not presented to the **Configuration Register File**, the SPI Flash Memory is not mapped to the memory address space.

The mapped page is selected by setting **Page Reg<2:0>** (**Our Register 1**).

The base address is defined in the **Expansion Rom Base Address Register**. The **Expansion Rom Base Address Register** also holds the page size and **ROMEN** which controls enabling of the expansion ROM.

The 4 kBytes sector (0x1F000 - 0x1FFFF) holds data configuring the network adapter after **PERSTn**.

In absence of the TWSI EEPROM the SPI Flash Memory contains also VPD Data (0x1C000, 0x1C0FF, **SPI Flash Memory VPD Configuration Register**).

Memory accesses to the SPI Flash Memory are read only.

Write operations are completed normally on the bus and the data is discarded.

For programming of the SPI Flash Memory no additional 12V power supply and switching of the programming voltage is required.

The SPI Flash Memory (256 kB) is organized in eight sectors with 32 kBytes each and 128 pages (page = 256 Byte) per sector:

Table 13: Eight SPI Flash Memory Sectors

Sector Address	Sector
000000 to 007FFF	1
008000 to 00FFFF	2
010000 to 017FFF	3
018000 to 01FFFF	4
020000 to 027FFF	5
028000 to 02FFFF	6
030000 to 037FFF	7
038000 to 03FFFF	8

2.4 SPI Flash Memory Loader

The SPI Flash Memory Loader supports the following features:

- Loading of data after **PERSTn** from the SPI Flash Memory into the Configuration and Control Register File (where needed).
  - The loader is capable of accessing potentially all registers in the **Control Register File** space.
  - Register address and data are stored in 8-byte entries in the SPI Flash Memory.
  - The registers may be written with dword, word or byte accesses.
  - The 8-byte entries are located on 8-byte boundaries starting at address 0x1F800 (or 0x1F000 for Plug-In-Go, configurable within **SPI Flash Memory Loader Control Register**) of the SPI Flash Memory in increasing order. Each entry is marked with a key.
    The selection of Plug-In-Go or normal operation configuration is done depending on the detected power supply by V_{AUX} or PCI power line.
  - If started, the loader reads subsequent entries starting with the initial value of the **Normal Loader Start Address** (Bit 27:16 in **SPI Flash Memory Loader Configuration Register**) or, if Plug In Go, with the value of the **PiG Loader Start Address** (Bit 11:0).
  - Loading is started by deassertion of **PERSTn** or the setting of the SPI **loader start** bit in the **SPI Flash Memory Control Register**.
  - While loading, accesses to any resource of the network adapter are terminated by Target Retry Cycles.
  - The transferred data after PERSTn in this way is limited to fulfill the requirements of PCI bus[1].
  - The command **SPI loader start** is intended for testing purposes only. It is not recommended to reload the **Configuration Register File** by using this command.

Table 14: First 8 byte unit within Normal Operation Configuration Region of SPI Flash Memory

31:24	23:16	15:8		7:0	Address
Address/upper	Address/lower	reserved	BE<3:0>	key = 0x55	0x1f800
Data<3>	Data<2>	Data<1>		Data<0>	0x1f804

- Loading Boot-Code:
  The boot data can be accessed as byte, word, or dword at a time. All combinations of byte-enables of the PCI specification are supported. There is a cache, which holds 2 consecutive dwords (8 bytes) of data. Initially, there is no valid data in the cache. When a boot access is initiated, consecutive dwords are fetched, starting with the dword containing the requested data. The requested byte/word/dword is forwarded as soon as the dword that contains the accessed address is loaded. The loader doesn't wait for the following dword fetch also to complete. A cache location (dword) is released (declared empty) when byte 4 in that dword is read out. Reading byte 4 may be done by dword/word/byte access. The SPI read continues as long as there is at least one empty location in the cache. This means that, in theory, the SPI memory could be transferred within one single burst throughout a sequential data load. The burst is broken when there is a jump in address or the access over the PCI bus takes too long time and the cache fills up. When requested data is in the cache (hit), data is supplied from the cache and no new SPI access is required. A new SPI access is initiated when there is a miss (requested data is not found in the cache or that dword is currently being fetched). The cache contents are made invalid when the SPI memory is written to.

- Loading VPD-Data:
  If the VPD-Data is in the SPI Flash Memory the whole 256 Byte can be accessed by 8 Read-Cycles to the VPD area (0x1c000 - 0x1c0ff). The data are stored in the lower dword of the cache.

1. T_{rhfa} PERSTn High to First configuration Access is limited to 2^{25} clock cycles.
PCI Express: 1s -> max. 2.8 KB

- Programming interface:
  The data can be written byte, word or dword at a time. All combinations of byte-enables of the PCI specification are supported. The starting byte could be anywhere within a page. When the end of the page is reached, the address wraps around to the beginning of the same page and the data will be stored there. This means that the Software is capable to handle the sector and page borders that no data are lost.
  Note: The same byte can't be reprogrammed without erasing the whole sector first.

- Support of SPI Flash Memories of different vendors:
  The current SPI Flash Memory devices of different vendors vary in their Instruction-Codes for the read ID instruction. After reset the used SPI Flash Memory needs to be identified by the SPI Flash Memory Loader. The identification of the Flash on the board after **PERSTn** should look like this:
  - after the release of **PERSTn** the SPI Flash Memory Loader starts directly to read the Configuration Data. If the SPI Flash Memory is programmed the loader reads all data into the Register-File until key (0x55) cannot be found anymore. Then and also if the SPI Flash Memory was not programmed at all the loader moves to an idle state to wait for new commands.
  - Afterwards the Software identifies the Flash. Therefore the Vendor- and Device-ID are read out of the SPI Flash Memory and the memory type is stored in the **RD ID Protocol** bit in **SPI Flash Memory Control Register**.
    Depending on the EPROM type the software uses different instruction code sets (**SPI Flash Memory Opcode 1 and 2 Register**).
    Even during identification of the SPI Flash Memory there are two possible protocols to be used. Maybe both protocols must be tried by software to read out the SPI Flash Memory ID.

- Sector Erase:
  Due to the fact that the Flash is divided into sectors only, a write to the VPD or configuration region leads to the erasure and rewriting of the whole VPD/configuration information.
  Reprogrammable VPD during operation of the device can be achieved by the use of an additional TWSI EEPROM (see chapter *2.5 TWSI EEPROM* on page 22).

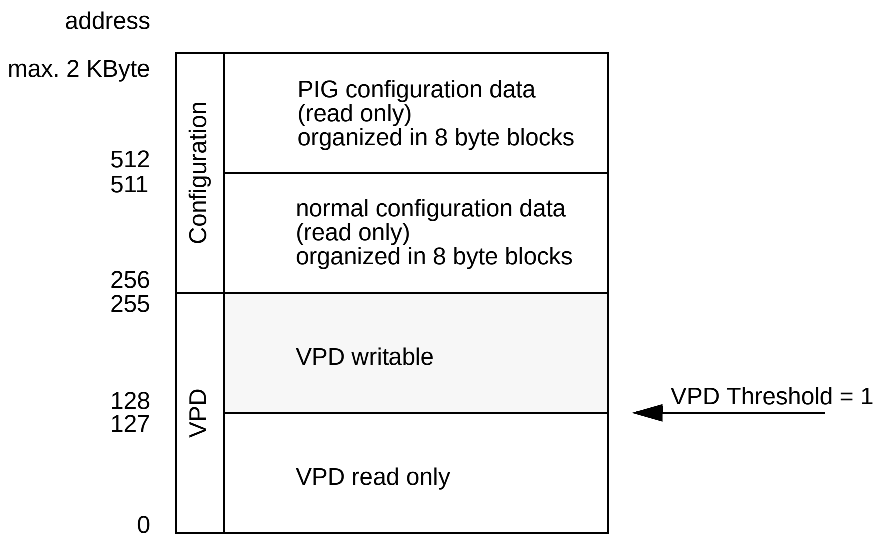
- Supported Instructions for the SPI Flash Memory:
  - Write - Program Data into Memory Array
  - Chip Erase - Erase All Sectors in Memory Array
  - Sector Erase - Erase One Sector in Memory Array
  - WREN - Set Write Enable Latch - has to be set before Program, Chip- and Sector-Erase
  - Read - Read Data from Memory Array
  - RDSR - Read Status Register
  - RDID - Read Vendor and Device ID (two different protocols)
  - NOP - No Operation

2.5 TWSI EEPROM

The TWSI EEPROM is an external memory device for VPD (vital product data).

Its address space is divided into two parts (see Figure 4).

Figure 4: Internal Structure of TWSI EEPROM.



Within the lower address region (0 to 255) the VPD is located. This part contains the read only and writable section of VPD separated by VPD Threshold (see **VPD Write Thr** in Our Register 2 in chapter *3.2.4.1 Our Register 1 and Our Register 2* on page 69).

In higher addresses the configuration data is stored which is loaded automatically into the ASIC after PERSTn with its internal TWSI EEPROM Loader (see chapter *2.6 TWSI EEPROM Loader* on page 23).
Two different configurations may be loaded by the TWSI EEPROM Loader depending on the current operation mode “normal” or “Plug in Go”.
Normally this feature is not used due to the existence of the SPI Flash Memory, which also holds configuration data.

The TWSI EEPROM is read and written to via TWSI bus. Its TWSI address is 0b101000.

The size of the TWSI EEPROM, the amount of writable VPD area and the Device Select Byte used for VPD TWSI accesses are determined by the read only fields **VPD ROM Size**, **VPD Write Threshold** and **VPD Devsel**[1] in **Our Register 2** (see chapter *3.2.4.1 Our Register 1 and Our Register 2* on page 69). These fields can be reloaded from the SPI Flash Memory, if another device is used.

For manufacturing programming of the read only part of the TWSI EEPROM, Testmode (**En Config Write**) must be set. Then the whole TWSI EEPROM is writable. Programming of the TWSI EEPROM is managed with ASIC internal registers **VPD Address** and **VPD Data** (see chapter *3.2.4.10 VPD Address Register* on page 78 and chapter *3.2.4.11 VPD Data Register* on page 78).

After the next power cycle the read only areas within the TWSI EEPROM are write protected again

1. Hint: **VPD Devsel must not be overwritten via TWSI EEPROM in absence of the SPI flash memory**. This may lead to a complete damage of the board (TWSI EEPROM must be changed afterwards!!!

2.6 TWSI EEPROM Loader

The TWSI EEPROM Loader looks for configuration data within the external TWSI EEPROM, although the configuration data is stored within the SPI Flash Memory and therefore no data is to be read out normally.

- The TWSI EEPROM Loader is active once after the execution of the SPI Flash Memory Loader (after PERSTn).
- It works analogously to the SPI Flash Memory Loader and loads startup data into the Configuration and Control Register File (where needed):
  - The loader is capable of accessing potentially all registers in the **Control Register File** space.
  - Register address and data are stored in 8-byte entries in the TWSI EEPROM (see Table 15 for details).
  - The 8-byte entries are located on 8-byte boundaries up from address 256 of the TWSI EEPROM in increasing order. Each entry is marked with a key byte (0x55).

Table 15: Data format of first 8 byte block within TWSI EEPROM

TWSI access cycle number	TWSI EEPROM Address	contents	
2	0x107	**Data<3>**	
	0x106	**Data<2>**	
	0x105	**Data<1>**	
	0x104	**Data<0>**	
1	0x103	**Address/upper**	
	0x102	**Address/lower**	
	0x101	**reserved**	**BE<0:3>**
	0x100	**Key = 0x55**	

  - The ASIC internal registers may be written with dword, word or byte accesses.
  - If started, the loader reads subsequent entries starting with the initial value of the **TWSI EEPROM Address Counter** (see *3.2.4.12 TWSI EEPROM Control Register* on page 79) as long as a valid key is found.
  - Loading is started by finishing the SPI Flash Memory Loader state machine or by setting the **Flag** to start the TWSI EEPROM Loader (see *3.2.4.12 TWSI EEPROM Control Register* on page 79).
  - Loading the TWSI EEPROM via Flag in the **TWSI EEPROM Control Register** is intended for testing purposes only. It is not recommended to reload the **Configuration Register File** using this command.
  - While loading, accesses to any resource of the network adapter are terminated by Target Retry Cycles.
  - The transferred data after PERSTn in this way is limited to fulfill the requirements of PCI bus[1].
  - This time consuming reading via TWSI bus may be deactivated by setting the **Flag** bit to *stop* (0) in the **TWSI EEPROM Control Register** see *3.2.4.12 TWSI EEPROM Control Register* on page 79. Then the state machine does not even start reading values.
- Transformation of the TWSI EEPROM data (8 bytes) into multiple byte/dword memory read accesses from the bus.
  - 32-bit read data is received via TWSI-Bus within one read access. Two read accesses are necessary to receive the full information for writing one internal register.
- Programming the TWSI EEPROM is described in chapter *2.5 TWSI EEPROM* on page 22

1. T_{rhfa} PERSTn High to First configuration Access is limited to 2^{25} clock cycles.
PCI Express: 1s -> max. 2.8 KB

2.7 Plug In Go Unit

The Plug In Go unit decides wether to load normal configuration or PiG configuration by the SPI Flash Memory Loader.

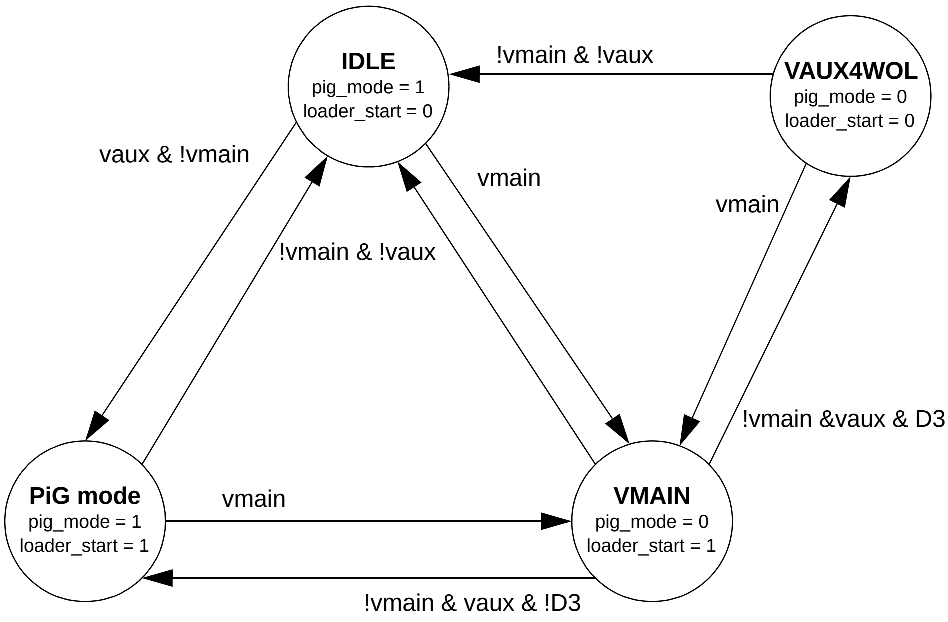
Which configuration to select is dependent on the available voltages V_{main} and V_{aux} and if the device is in the D3 state.

The Plug in Go configuration is loaded by the SPI Flash Memory loader when V_{main} is no longer available but V_{aux} and the device is not in D3 state.

After a power on (both voltages not available) PiG configuration is loaded when only V_{aux} available. Else normal configuration is loaded by the SPI Flash Memory loader.

See figure below for the complete state description of the Plug in Go unit:

Figure 5: Plug in Go Unit State Diagram



loader_start = 1 causes the SPI Flash Memory loader to load the configuration once after entering the state

pig_mode = 0: normal configuration is loaded by the SPI Flash Memory loader
pig_mode = 1: Plug in Go configuration is loaded by the SPI Flash Memory loader

2.8 Interrupts

Chip internal interrupt handling is done with the following internal signals:

- **irq** is asserted when at least one of the unmasked interrupt sources is active (interrupt).
  When **irq** is not asserted, then none of the unmasked interrupt sources is active **or** internal interrupt masking line **isr_mask** is active.
- **isr_status** is asserted (set to ‘1’) when entering the interrupt service routine: “ISR mode”.
  isr_status is deasserted (set to ‘0’) when leaving the interrupt service routine: “normal mode”.
- **isr_mask** shows if currently all interrupts are masked (regardless of interrupt mask register settings).

The two internal signals irq and isr_status show the current state of interrupt processing:

Table 16: Interrupt Processing Signals

irq	isr_status	interrupt state	comment
0	0	idle	No interrupt request is pending
1	0	request	Interrupt pending, but not yet in process
0	1	processing	Interrupt is served but not yet finished. All further interrupts masked

isr_status is used by the status BMU to accelerate the processing according to the threshold settings for the status FIFO during interrupt processing.

The **Interrupt Source Register** holds the interrupts of all resources (see chapter *3.3.2.4 Interrupt Source Register* on page 130).
Each interrupt is maskable by the **Interrupt Mask Register** (see chapter *3.3.2.5 Interrupt Mask Register* on page 131).
All unmasked interrupts are Or’ed and propagated to the internal interrupt line **irq**.

Interrupts generated by hardware checks are readable via the **Interrupt Hardware Error Source Register** (see chapter *3.3.2.6 Interrupt HW Error Source Register* on page 131). Each interrupt is maskable by the **Interrupt Hardware Error Mask Register** (see chapter *3.3.2.7 Interrupt HW Error Mask Register* on page 132). All unmasked interrupts are Or’ed and propagated to the **Interrupt Source Register** as **Interrupt Hardware Error.**

The interrupts from the MAC are readable from the **MAC Interrupt Source Registers** (see chapter *3.3.2.54 MAC Interrupt Source Register* on page 201)**.** Each interrupt is maskable by the **MAC Interrupt Mask Registers** (see chapter *3.3.2.55 MAC Interrupt Mask Register* on page 202). All unmasked interrupts are Or’ed and propagated to the **Interrupt Source Register** as **MAC Interrupt.**

An interrupt from a masked source can still be read from its **Source Register**.
An interrupt is cleared and/or disabled as stated in the description of the related interrupt resource.

The **Special Interrupt Source Registers** mirror the **Interrupt Source Register** with special functionality adapted to typical SW handling:

- Special Interrupt Source Register 1:
  If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.
  If the internal interrupt line **irq** is NOT asserted, the read value is 0.
  If the internal interrupt line **irq** is asserted, reading the **Special Interrupt Source Register 1** masks all interrupts. As a result the internal interrupt line **irq** is deasserted.
- Special Interrupt Source Register 2:
  If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.
  If the internal interrupt line **irq** is NOT asserted, the read value is 0.
  If the internal interrupt line **irq** is asserted, reading the **Special Interrupt Source Register 2** masks all interrupts and sets **isr_status** flag to “ISR mode”. As a result the internal interrupt line **irq** is deasserted.
- Special Interrupt Source Register 3:
  If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.
  If the internal interrupt line **irq** is NOT asserted, the read value is 0.
  Reading the **Special Interrupt Source Register 3** always masks all interrupts. As a result the internal interrupt line **irq** is deasserted.
  If the internal interrupt line **irq** is asserted, **isr_status** flag is set to “ISR mode”.

Table 17: Effects on reading the Special Interrupt Source Registers

	Special Interrupt Source Register 1	**Special Interrupt Source Register 2**	**Special Interrupt Source Register 3**
Read value when **irq** asserted	Interrupt Source Register	Interrupt Source Register	Interrupt Source Register
Read value when **irq** not asserted	0	0	0
isr_status set to “ISR mode”	-	when **irq** is asserted	when **irq** is asserted
Masking of all interrupts	only, when **irq** is asserted	only, when **irq** is asserted	always
internal **irq** line	deasserted afterwards	deasserted afterwards	deasserted afterwards

All interrupts can be moderated by the **IRQ Moderation Timer**.

Moderation is controllable individually for each interrupt by the **Interrupt Moderation Mask Registers.**

There are two modes of signaling interrupts (internal interrupt line **irq** is active) to the host system:

- Via Interrupt line INTAn: The internal signal **irq** is forwarded to the PCI signal **INTA#** (unless inhibited by register setting).

- Via MSI (Message Signaled Interrupt):
  The MSI agent notifies the interrupt via busmaster write to the defined host memory address.
  When leaving the ISR and all interrupt sources have been cleared, the MSI agent is in state idle and waits for new interrupts. If there are still active interrupt sources or a new interrupt source is set to active, **irq** is asserted again and the MSI agent is triggered for a new interrupt request.

The mode is selected with **MSI Enable** bit in **MSI Message Control** register.

2.8.1 IRQ Moderation Timer

The **IRQ Moderation Timer** (see chapter *3.3.2.29 IRQ Moderation Timer Registers* on page 147) is a programmable 32-bit downcounter with a resolution of one core clock cycle (6.4 ns, T_{max} = 21.47 s) for the usage as time-base for IRQ Moderation.

The command **Interrupt Moderation Timer Start** loads **Interrupt Moderation Timer** with **Interrupt Moderation Timer Init Value** and starts counting down.

Reaching ZERO or loaded with ZERO the **Interrupt Moderation Timer** is reloaded with **Interrupt Moderation Timer Init Value**.

The **Interrupt Moderation Timer** controls the assertion of the internal interrupt line **irq** by gating the interrupts as defined by the **Interrupt Moderation Mask Registers**. If the **Interrupt Moderation Timer** is stopped or reaches ZERO, the gate opens and allows the masked interrupts to propagate to the bus. The assertion of **irq** has been caused by one of the masked interrupts, therefore delayed until the next time, the **Interrupt Moderation Timer** reaches ZERO. **irq** is kept asserted until the appropriate operation of clearing the interrupt request is completed. The deassertion of **irq** is not affected.

The **Interrupt Moderation Timer** may be stopped by the command **Interrupt Moderation Timer STOP**.

While **HW Reset** or **SW Reset** the **Interrupt Moderation Timer** is stopped and the gate is closed. After releasing **SW Reset** the gate is initially open until the **Interrupt Moderation Timer** is loaded with a value other than ZERO and started.

2.8.2 Message Signaled Interrupts (MSI)

Reporting interrupts to the host system via MSI is done, when the **MSI Enable** bit is set in **MSI Message Control** Register. Interrupt line INTA# is not used in this mode.

This chip is capable of handling one MSI message. This is specified also in the **MSI control register**.

The location for the MSI message is defined in register **MSI Message Address Upper/Lower** (64 bit). At this location within the host memory the interrupt message of the chip is written to.

The message itself is stored within register **MSI message data register**.

The MSI agent detects an active internal interrupt line **irq** and starts a busmaster write to the defined **MSI address** with the defined **MSI message data**.
The host system now detects a MSI message at the defined location and starts an interrupt service routine.

During the ISR all interrupts are masked.

After completion of the ISR the interrupts are unmasked again and the **irq** line is released in case of solving all interrupt reasons during the ISR. Otherwise the interrupt is signaled again to the host.

2.9 Buffer Management Units (BMU)

The Buffer Management Units are the interface for the Bus Interface Unit (BIU) to the queues.

The internal requests are presented as one for each queue.

Guaranteed length of transferable data is derived from the data provided by the BMU and the FIFO.

The Master Backends are also providing the multiplexers for positioning of data words at the right byte lanes on misaligned transfers and to revert byte ordering for descriptor words depending on **Rev Bytes Desc**.

The position of the multiplexers are controlled by the BMUs.

2.9.1 Format of Descriptor and Status List Elements

Communication between host and PCI device is done via interchanged list elements provided by the host within its memory space.
Chip internal descriptors can be modified via list elements and control functions are initiated or status information is reported.

All list elements of the different agents are based on the same structure (see figure below):

- Width of the list elements is always 64 bit
- Bit 63 is the own bit: it marks the ownership of the list element:
  1: the PCI device is owner of the list element
  0: the host is owner of the list element
- The following 7 bits (bit 62... bit 56) contain the opcode for this list element.
  The opcode defines the meaning of the remaining fields of the list element.
  The different opcodes are assigned to the different tasks as follows:

Table 18: Opcode Assignments

Opcode (hex)	Task group
0x00 - 0x0f	not defined
0x10 - 0x1f	TCP Sum Parameter or Function
0x20 - 0x2f	Register Updates for RX and TX
0x30 - 0x3f	not defined
0x40 - 0x4f	DMA for RX and TX
0x50 - 0x5f	not defined
0x60 - 0x6f	Status List Functions
0x70 - 0x7f	Special Actions (e.g. put index)

- The control field (bit 55.. bit 48) specifies additional attributes of the list element
- The remaining 48 bits (bit 47.. bit 0) hold data according to the opcode and control information.

Figure 6: Base Format of the List Elements

63	62 56	55 48	47	32	31			0
Own	Opcode	Control	data	data	data	data	data	data

The list elements of each task group are described in detail within the according chapters.

2.9.1.1 Receive Descriptor List Element

Figure 7 shows all different list elements processable by the RX BMU.

Figure 7: RX Descriptor List Element Definition

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **Packet**	control	buffer length	buffer address low dword

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **Buffer**	control	buffer length	buffer address low dword

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **ADDR64**	reserved	reserved	buffer address high dword

63	62 – 56	55 – 48	47 – 32	31 – 0	
Own	Opcode **TCPPAR**	reserved	reserved	TCP Start Sum 1	TCP Start Sum 2

The following opcodes are valid for RX descriptor list elements:

Table 19: RX descriptor list element valid opcodes

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x12	TCPPAR	Bits 31:0: TCP Sum Start Address 1 and 2
0x21	ADDR64	Bits 31:0: New value for the Buffer Address High Register of the RX BMU
0x40	Buffer	Follow up buffer for a packet Bits 55:48: buffer control bits Bits 47:32: buffer length Bits 31:0: buffer address low One buffer list element must be skipped on the start of a new packet. It enables a DMA request by the RX BMU to write receive data.
0x41	Packet	First buffer of a packet Bits 55:48: buffer control bits Bits 47:32: buffer length Bits 31:0: buffer address low A packet list element enables a DMA request by the RX BMU to write receive data.
other	not supported by RX BMU	Processing of list elements is stopped and an interrupt IRQ is asserted by BMU.

The Own bit signals the ownership of the list element:

- Own = 1: The PCI device is owner of the list element
- Own = 0: If Own = 0 in a list element detected by the BMU, it stops processing list elements and asserts an interrupt request IRQ.

The Control field defines attributes to the descriptor:

Table 20: Control Field Definitions

bit #	Control bit	Meaning (when set to 1)
7	**reserved**	
6	**FRC_STAT**	List element forces a burst of the Status FIFO
5:2	**reserved**	
1	**CALSUM**	Calculate checksum for this packet
0	**reserved**	

2.9.1.2 Transmit Descriptor List Element

Figure 8 shows all different list elements processable by the TX BMU. If not mentioned explicitly the ordering is little endian.

Figure 8: TX Descriptor List Element Definition

63	62–56	55–48	47–32	31–0
Own	Opcode **Packet**	Control	buffer length	buffer address low dword

63	62–56	55–48	47–32	31–0
Own	Opcode **Large Send**	Control	buffer length	buffer address low dword

63	62–56	55–48	47–32	31–0
Own	Opcode **Buffer**	Control	buffer length	buffer address low dword

63	62–56	55–48	47–32	31–0
Own	Opcode **ADDR64**	reserved	reserved	buffer address high dword

63	62–56	55–48	47–32	31–	–0
Own	Opcode **LRGLEN**	reserved	reserved	reserved	large send length

63	62–56	55–48	47–32	31–0
Own	Opcode **VLAN**	reserved	VLAN (big endian)	reserved

63	62–56	55–48	47–32	31–0
Own	Opcode **ADDR64 + VLAN**	reserved	VLAN (big endian)	buffer address high dword

63	62–56	55–48	47–32	31–	–0
Own	Opcode **LRGLEN + VLAN**	reserved	VLAN (big endian)	reserved	large send segment length

63	62–56	55–48	47–32	31–	–0
Own	Opcode **TCP LISW***	lock number	TCP sum init value (big endian)	TCP sum start	TCP sum write

***L, I, S, W stand for the fields. Not all are used.**

The following opcodes are valid for TX descriptor list elements:

Table 21: Valid Opcodes for TX descriptor list elements

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x12	TCP __S_	Load value from list element: Bits 31:16: TCP sum start address
0x16	TCP _IS_	Load value from list element: Bits 47:32: TCP sum init value Bits 31:16: TCP sum start address The byte order within the TCP sum init value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).
0x18	TCP L___	Bits 55:48: lock this number of packets Set TCP lock for number of packets.
0x19	TCP L__W	Bits 55:48: lock this number of packets Load value from list element: Bits 15:0: TCP sum write address Write TCP sum into this packet.
0x1B	TCP L_SW	Bits 55:48: lock this number of packets Load values from list element: Bits 31:16: TCP sum start address Bits 15:0: TCP sum write address Write TCP sum into this packet.
0x1F	TCP LISW	Bits 55:48: lock this number of packets. Load values from list element: Bits 47:32: TCP sum init value Bits 31:16: TCP sum start address Bits 15:0: TCP sum write address Write TCP sum into this packet. There are several TCP Parameter Opcodes, but not all combinations are valid. Valid combinations: TCP LISW, TCP L_SW, TCP L__W, TCP L___, TCP _IS_, TCP __S_. The byte order within the TCP sum init value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).
0x21	ADDR64	Bits 31:0: New value for the Buffer Address High Register of the TX BMU

Table 21: Valid Opcodes for TX descriptor list elements

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x22	VLAN	Bits 47:32: New value for VLAN Tag Register of the TX BMU The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x23	ADDR64 + VLAN	Bits 47:32: New value for VLAN Tag Register of the TX BMU Bits 31:0: New value for the Buffer Address High Register of the TX BMU The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x24	LRGLEN	Bits 15:0: MTU for TCP Segmentation
0x26	LRGLEN + VLAN	Bits 47:32: new value for VLAN Tag Register of the TX BMU Bits 15:0: MTU for TCP Segmentation The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x40	Buffer	Follow up buffer for a large send or normal packet Bits 55:48: buffer control bits Bits 47:32: buffer length Bits 31:0: buffer address low It triggers a DMA request by the TX BMU to read transmit data.
0x41	Packet	First buffer of a normal packet: Bits 55:48: buffer control bits Bits 47:32: buffer length Bits 31:0: buffer address low A packet list element enables a DMA request by the TX BMU to read transmit data.
0x43	Large Send	First buffer of a large send packet: Bits 55:48: buffer control bits Bits 47:32: buffer length Bits 31:0: buffer address low A packet list element enables a DMA request by the TX BMU to read transmit data.
other	not supported by TX BMU	Processing of list elements is stopped and an interrupt IRQ is asserted by BMU.

The Own bit signals the ownership of the list element:

- Own = 1: The PCI device is owner of the list element
- Own = 0: If Own = 0 of a list element detected by the BMU, it stops processing list elements and asserts an interrupt request IRQ.

The Control field defines attributes to the descriptor:

Table 22: Control Field definition attributes to the descriptor

bit #	Control bit	Meaning (when set to 1)
7	**EOP**	Descriptor is last one of a packet
6	**FRC_STAT**	Descriptor forces a TX Status Element in the Status FIFO
5	**INS_VLAN**	Insertion of VLAN Tag from TX VLAN Tag Register into packet
4	**LOCKSUM**	Start of lock packet sequence (number of packets stored in lock register). To be set only for the first packet of a lock sequence to accept value from internal lock register (lock number value transmitted with last TCP_LISW list element). The number of locked packets is put in the Rambuffer subsequently.
3	**INITSUM**	Start checksum calculation new for this packet and use value of init register for initialization of checksum calculation.
2	**WRITESUM**	Write checksum value to this packet at write position.
1	**CALSUM**	Calculate checksum for this packet
0	**UDPTCP**	1: Calculate UDP checksum for this packet 0: Calculate TCP checksum for this packet

2.9.1.3 Status List Element

Figure 9 shows all different list elements processable by the Status BMU.

Figure 9: Status List Element Definition

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **RX Status**	Link	RX Frame Length little endian	RX Frame Status Word little endian

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **RX Time-stamp**	Link	(VLAN Tag) big endian	RX Frame Timestamp little endian

63	62 – 56	55 – 48	47 – 32	31 – 0
Own	Opcode **RX VLAN**	Link	VLAN Tag big endian	reserved

63	62 – 56	55 – 48	47 – 32	31 – 0	
Own	Opcode **TCP Sum (+VLAN)**	Link	(VLAN Tag) big endian	TCP sum 2 big endian	TCP sum 1 big endian

63	62 – 56	55 – 48	47 – 32			31 – 0
Own	Opcode **RSS Hash**	Link	reserved	TCP	IP	RSS Hash value little endian

63	62 – 56	55 – 48	47 – 32		31 – 0	
Own	Opcode **TX Index**	reserved	TXS2 Done	TXA2 Done	TXS1 Done	TXA1 Done

The following opcodes are valid for Status list elements:

Table 23: Valid Opcodes for Status List Elements

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x60	RX Status	Generated by the RX BMU for each packet: Bits 55:48: Link number Bits 47:32: RX frame length Bits 31:0: RX status word
0x61	RX Timestamp	Generated by the RX BMU for each non-VLAN packet, if the timestamp timer is enabled and RSS disabled[1]: Bits 55:48: Link number Bits 31:0: RX timestamp
0x62	RX VLAN	Generated by the RX BMU for each VLAN packet and the timestamp timer and check summing is disabled: Bits 55:48: Link number Bits 47:32: VLAN Tag The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x63	RX Timestamp + RX VLAN	Generated by the RX BMU for each VLAN packet and the timestamp timer is enabled and RSS is disabled: Bits 55:48: Link number Bits 47:32: VLAN Tag Bits 31:0: RX timestamp The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x64	TCP Sum	Generated by the RX BMU for each non VLAN packet, if checksumming is enabled Bits 55:48: Link number Bits 31:16: TCP Sum 2 Bits 15:0: TCP Sum 1 The byte order within the TCP Sum is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).
0x65	RX RSS Hash	Generated by the RX BMU for each packet, if RSS Hash calculation is enabled) Bits 55:48: Link number Bit 33: TCP Flag Bit 32: IP Flag Bits 31:0: RSS Hash value

Table 23: Valid Opcodes for Status List Elements

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x66	TCP Sum + VLAN	Generated by the RX BMU for each VLAN packet, if check-summing is enabled Bits 55:48: Link number Bits 47:32: VLAN Tag Bits 31:16: TCP Sum 2 Bits 15:0: TCP Sum 1 The byte order within the TCP Sum is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8). The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 39:32) and the LSB is in the higher byte (bit 47:40).
0x68	TX Index	If an Status Burst is scheduled and at least one of the TX Done indices differs from its TX Report Index, a TX Index LE is appended as last LE in the Status Burst. Bits 31:25: TXS2 Done Index Bits 24:16: TXA2 Done Index Bits 15:8: TXS1 Done Index Bits 7:0: TXA1 Done Index
other	not supported by Status BMU	

1. Note, that RX RSS Hash LE and Timestamp LE can only be generated exclusively. As long as the RSS Hash feature is enabled, no Timestamp LE is generated.

2.9.1.4 Special Action List Elements

Figure 10 shows all different special action list elements.

Figure 10: Special Action List Elements

63	62 – 56	55 – 48	47 – 44	43 – 32	31 – 28	27 – 16	15 – 12	11 – 0
Own	Opcode **Put Index**	Link	res.	RX Put Index	res.		res.	TXA Put Index

The following opcodes are valid for special action list elements:

Table 24: Valid Opcodes for Special Action List Elements

Opcode (hex) 7 bit	Opcode Name	Description of assigned list element
0x70	Put Index	List element for Hardware Polling Unit: Bits 55:48: Link number Bits 47:44: reserved Bits 43:32: RX Put Index Bits 31:28: reserved Bits 27:16: TXS Put Index Bits 15:12: reserved Bits 11:0: TXA Put Index
other	not defined yet	

2.9.2 TCP/UDP Processing of RX and TX BMU

TCP checksum may be calculated by RX and TX BMU for the processed packet.

2.9.2.1 TCP checksum in RX direction

In RX direction two TCP Checksums may be calculated from two different configurable start positions (refer to 3.3.2.41 *BMU Registers for Receive Queues* on page 158 and 2.9.1.1 *Receive Descriptor List Element* on page 30).

2.9.2.2 TCP/UDP checksum for single packets in TX direction

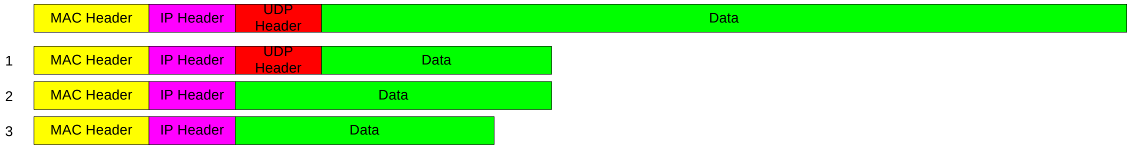
In TX direction TCP or UDP checksum may be calculated.

The control bit UDPTCP must be set accordingly. Init Value, Start address and Write address are defined for the TCP/UDP Sum. Store and Forward (St&Fwd) and Calculate TCP/UDP Sum for this buffer/packet must be set.

2.9.2.3 UDP checksum for fragmented packets in TX direction

When large UDP packets are split into several smaller packets. UDP Checksum is calculated for the original large packet. Therefore the UDP is calculated over the data areas of several small packets and the resulting checksum is inserted within the UDP header field of the first packet. The address for the insertion of the checksum is loaded explicitly and is not updated automatically by a new packet start.

Figure 11: UDP checksum across several packages



The software can handle the several packets belonging to one UDP checksum processing in two order modes. The packets are handled in the original order to the hardware or they are transferred starting with packet number 2 up to the last of the sequence and terminated by the first one.

UDP processing for packets in original order (1, 2,..., n):

- Set TCP Lock for n packets, which inhibits transmission of the next n packets coming in.
- Set TCP Sum Write: remember the RAM Buffer Address, where to write the TCP Sum.
- Set TCP Sum Init: initialize the Checksum Adder with a start value.
- Set TCP Sum Start: set the start address for checksum calculation for the 1st packet.
- Buffer with 1st packet: calculate checksum with current parameters.
- Set TCP Sum Start: set the start address for checksum calculation for the 2nd and following packets.
- Buffer with 2nd and following packets: calculate checksum with current parameters.
- EOF of last packet clears TCP Lock, writes checksum to first packet and allows transmission of all packets.

In this case the packets can be sent after the checksum was calculated including the last packet.

UDP processing for packets in "Vendel" order (2,..., n, 1):

- Set TCP Sum Init: initialize the Checksum Adder with a start value, when control bit initsum is set to '1' for the current packet.
- Set TCP Sum Start: set the start address for checksum calculation for the 2nd and following packets.
- Buffer with 2nd and following packets: calculate checksum with current parameters.
- Packets are sent immediately after complete reception (2.. n).
- Set TCP Sum Write: remember the RAM Buffer Address, where to write the TCP Sum. For the last transmitted packet (number 1) the control bit writesum must be set to '1'.
- Set TCP Sum Start: set the start address for checksum calculation for the 1st packet.
- Buffer with 1st packet: calculate checksum with current parameters.
- EOF of packet 1 clears TCP Lock, writes checksum to packet 1 and allows transmission of packet 1.

In the "Vendel" case the packets can be sent just after receipt and the calculated checksum is inserted in packet 1, which is received and transmitted last.

2.9.3 Prefetch Unit

Each RX or TX FIFO has its own prefetch FIFO. This block occurs two times per link:

- RX Prefetch Unit
- TX Prefetch Unit

The Prefetch Unit does intelligent prefetching of list elements for a BMU.

The host software may update the according Put Index Register continuously to trigger the prefetch of new list elements completely automatically.

Alternatively the Put Index value is polled continuously by the HW Polling Unit out of a software defined memory location and the Put Index Register of the Prefetch Unit is updated by the HW Polling Unit.

A prefetch of new list elements is started whenever Put Index Register differs from Get Index Register and there is at least the minimum free space left in the Prefetch FIFO. The number of new list elements which are to be prefetched is calculated (Put Index - Get Index and minimum free space of Prefetch FIFO).

The Prefetch FIFO is written by the BIU and read by the BMU.

It works fully synchronously to the BIU Master Interface and the associated BMU.

It is controlled by the following parameters and variables:

Table 25: Prefetch Control Parameter and Variables

Name	Description
List Start Address Low & High	64 bit pointer to the beginning of the Descriptor List within the host buffer. Defined by software during queue initialization.
List Length	Size of the Descriptor List area in bytes. Defined by software during queue initialization.
Get Index	Pointer to the next list element, which is to be prefetched. Defined by software during queue initialization. Updated automatically on each prefetch burst.
Get Length	Number of list elements which may be prefetched by the next prefetch burst. Number = Put Index - Get Index Limitation: a guaranteed minimum space must be left free within the prefetch FIFO.
Put Index	Pointer to the last list element put by host software. Either written by host software each time new list elements haven been added to the descriptor list, or automatically updated by a Put Index Poll Request in HW Polling Mode.
FIFO Read Pointer	Read Address for FIFO memory Updated automatically when BMU proceeds to the next list element.
FIFO Write Pointer	Write Address for FIFO memory Updated automatically on each prefetch burst. Related closely to the Get Index.
FIFO level	FIFO filling level. Base for trigger condition for polling and prefetch bursts.

2.9.4 Status BMU

The status BMU handles information flow between hardware and host software.

This is done with a status list area within the host memory space. This area is provided by the software and must be large enough for the reception of status information about a number of packets for the two receive and transmit queues. There is no hardware mechanism to prevent the Status BMU from overwriting previous status list elements before they are processed by the host software.

The status list is write only for the hardware.

The host software resets the own bits of the list elements it has already processed to prevent wrapping around the list.

All RX Status list elements concerning the same packet are kept together. They are put to the list in the following order (ascending opcode):

- RX Checksum (+VLAN) if checksumming is enabled
- RX VLAN, if VLAN packet and previous LEs did not comprise VLAN.
- RX Status: The list element holding RX status word is always the last within this block.

The Status BMU is controlled by the following parameters and variables:

Table 26: Status BMU Control Parameters and Variables

Name	Description
List Start Address Low & High	64 bit pointer to the beginning of the Descriptor List within the host buffer. Defined by software during queue initialization.
List Length	Size of the Descriptor List area in bytes. Defined by software during queue initialization.
TXA Report Index	Last reported TX done index for the transmit queues
TX Index Threshold	Threshold for initiating a status burst
Put Index	Pointer to the next free list element in the host memory descriptor list area. If Put Index and Get Index are equal no element is in the FIFO.
FIFO Read Pointer	Read Address for FIFO memory Updated automatically on each status burst.
FIFO Write Pointer	Write Address for FIFO memory
FIFO watermark	FIFO watermark for initiating a status burst.
FIFO ISR watermark	FIFO watermark for initiating a status burst during ISR.
Level Timer	If the status FIFO is not empty the Level Timer is started. When expired a status burst is triggered
TX Timer	The timer is started when the TX Done Index differs from its TX Request Index. When expired a status burst is triggered.
ISR Timer	During ISR the ISR Timer is used instead of the Level Timer.

If one of the TX Done conditions is fulfilled, a TX Index list element is appended as last list element in the Status Burst.

A burst to the status list affects the following registers and timers:

- TX Report Index = TX Done Index for the TX queues

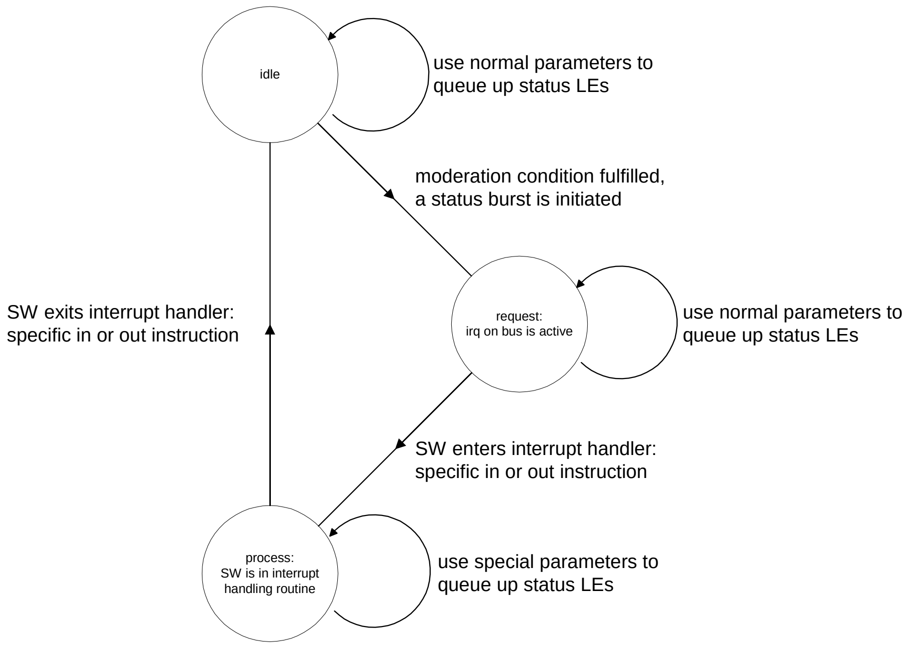
After the burst to the status list an interrupt Status List is asserted.
Bit **Status BMU** in **Interrupt Source Register** is asserted.

If new list elements reach the Status FIFO before software starts interrupt processing, they are queued up in the FIFO. As soon as software starts interrupt processing (e.g. clears IRQ Mask register) the Status FIFO uses other moderation parameters: The Level Timer is replaced by an ISR Timer, that is set by software to expire shortly before the expected leaving of the ISR (Interrupt Service Routine). The watermark is replaced by the ISR watermark, which triggers bursts earlier while software is in the ISR. This way the Status BMU FIFO is kept empty and all queued and newly incoming list elements are forwarded to the BIU as soon as possible.

The TX Done Index of the TX queue is connected to the Status BMU. The Status BMU maintains a TX Report Index for the TX queue holding the last state of the TX Done Index last reported to the host.
A TX Index Timer is set to its initial value and started each time one of the TX Done Index differs from its TX Report Index. It is stopped and reset to ZERO when the TX Report Index is updated.
Each time one of the Done Index differs from its Report Index and the difference is higher than the TX Index Threshold or the TX Timer expires, a Status Burst request is asserted to the Status BMU.

During the processing of a Status burst, new list elements may be queued up and they are also sent within an immediately following burst (burst size and timer values are lower than for normal burst requests; see Figure 12: *Status Interrupt Moderation*). The TX Index list element will be appended to the last element of the last status burst. Right before being appended the states of the TX Done Index is sampled and inserted into the TX Index list element and the TX Done Index is copied to the according TX Report Index.

Figure 12: Status Interrupt Moderation



2.9.5 Polling Unit

The Polling Unit is used to automatically update the Put Index Registers of the BMUs.
It runs synchronously to the BIU Master Interface and all BMUs.
It is a read only agent on the BIU's request interface.

The Put Indices and their enable bits for all BMU descriptor lists are stored in a contiguous memory area within the host buffer which is formally organized as two list elements.

The Polling Unit reads always both list elements at once and the values are stored in an internal register bank.
If the Enable bit of the Put Index is set, the value is forwarded directly to the according BMU without using the target bus.

Each time the Descriptor Poll Timer expires a Poll Request is initiated.
The Descriptor Poll Timer runs with clk_core to have a fixed timebase.

For testing a Poll Request can be started by software via Poll Control Register.

The Polling Unit is controlled by the following parameters and variables:

Table 27: Polling Unit Control Parameters and Variables

Name	Description
Poll Start Address Low & High	64 bit pointer to the beginning of the Descriptor List within the host buffer. Defined by software during Polling Unit initialization.
Last Index	Index of the last list element to poll. Fixed by hardware to 1. Two list elements are read always.
Poll Control	Control bits for Polling Unit: Reset/Enable Operational On/Off do_poll: command for polling

A check interrupt is generated on one of the following reasons:

- the received list element has the OWN bit (bit 63) not set.
- the received list element has a wrong opcode.
- at least one byte enable is not set for the received list element

The interrupt is cleared by software with setting the Clear IRQ bit in the Poll Control Register

2.10 Timer

The **Timer** is a programmable 32-bit downcounter with a resolution of 8 ns (derived from core clock) for the usage as a fixed timebase.

The command **Timer Start** loads **Timer** with **Timer Init Value** and starts counting.

Reaching ZERO or loaded with ZERO the **Timer** generates an interrupt **IRQ Timer** and is reloaded with **Timer Init Value**.

IRQ Timer is cleared by the command **Timer Clear IRQ**.

Command **Timer Clear IRQ** overrides a concurrent internal interrupt (guaranteeing IRQ edges).

Note

In order to prevent IRQ pulses, command **Timer Clear IRQ** should only be issued, if **IRQ Timer** is pending or if the **Timer** is stopped.

The **Timer** may be stopped by the command **Timer STOP**.

Testing:

Test mode is switched on/off by the command **Timer Test On/Off**. In test mode, clock pulses may be generated by software command **Timer Step**.

2.11 Timestamp Timer

The **Timestamp Timer** is a programmable 32-bit up counter with a resolution of 8 ns (derived from core clock) for the usage as a time base for time stamping the receive frames.

The command **Timestamp Timer Start** starts counting (from its current value).
Wrapping from 0xffffffff to ZERO or loaded with ZERO, the **Timestamp Timer** generates an interrupt **IRQ Timestamp Timer**.
IRQ Timestamp Timer is cleared by the command **Timestamp Timer Clear IRQ**. Command **Timestamp Timer Clear IRQ** overrides a concurrent internal interrupt (guaranteeing IRQ edges).
The **Timestamp Timer** may be stopped by the command **Timestamp Timer STOP**.

Testing:

Test mode is switched on/off by the command **Timestamp Timer Test On/Off**.
In test mode, clock pulses may be generated by software command **Timestamp Timer Step**.

2.12 Wake on LAN

Via PME line the host system can be woken by the PCI device, when a wake up event is detected on the LAN interface. This may be done for all three operation modes 10/100/1000.

The Wake on LAN feature uses three mechanisms to create a wake up event:

- Wake up Frame: Incoming packets are compared to several patterns stored in a RAM. A match causes a wake up event.
  Magic Packet frame detect: The incoming data stream is searched for a so called “magic packet frame” that consists of 6 bytes of 0xFF followed by 16 iterations of the device’s MAC address. If this sequence is found, a wake up event is created.
- Link Change monitoring: Any change of the link status causes a wake up event.

2.12.1 Wake up Frame Logic

The Wake up Frame Logic consists of a Pattern RAM, a receive data register, length counters and compare logic for each pattern to compare and a statemachine that parses the incoming data stream for start and end of packet. A control register holds bits to enable and configure the Wake up Frame logic and a status register holds bits that show the result of the matching process.

The complete set of available registers is described in chapter *3.3.2.57 Wake on LAN Control Registers* on page 203. The representation of the pattern RAM to the software can be found in chapter *3.3.2.58 Pattern RAM* on page 208.

The Wake up Frame Logic is run with the MAC’s receive clock (same as the MAC RX FIFO).

The Pattern RAM is a Single Port SRAM of 64 words by 128 bits. Under WOL working condition the RAM port is used by the compare logic to read out the patterns for comparison. To access the RAM from the PCI bus by target read or target write to set up the patterns the WOL unit must be set to inactive before (bit 1:0 to 0b01 of WOL Control Register).

Figure 13: Organization of Pattern RAM

Byte # /addr.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	mask byte 1	pat 6, byte 1	pat 5, byte 1	pat 4, byte 1	pat 3, byte 1	pat 2, byte 1	pat 1, byte 1	pat 0, byte 1	mask byte 0	pat 6, byte 0	pat 5, byte 0	pat 4, byte 0	pat 3, byte 0	pat 2, byte 0	pat 1, byte 0	pat 0, byte 0
0x01	mask byte 3	pat 6, byte 3	pat 5, byte 3	pat 4, byte 3	pat 3, byte 3	pat 2, byte 3	pat 1, byte 3	pat 0, byte 3	mask byte 2	pat 6, byte 2	pat 5, byte 2	pat 4, byte 2	pat 3, byte 2	pat 2, byte 2	pat 1, byte 2	pat 0, byte 2
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
0x3f	mask byte 127	pat 6, byte 127	pat 5, byte 127	pat 4, byte 127	pat 3, byte 127	pat 2, byte 127	pat 1, byte 127	pat 0, byte 127	mask byte 126	pat 6, byte 126	pat 5, byte 126	pat 4, byte 126	pat 3, byte 126	pat 2, byte 126	pat 1, byte 126	pat 0, byte 126

The Pattern RAM holds 7 Patterns of up to 128 Bytes each. Each data word consists of two bytes of each Pattern on byte 0 .. 6 and byte 8 .. 14, while bit 0..6 of byte 7 and byte 15 are used as mask bits to enable the comparison of the current byte for each pattern. Bit 0 of byte 7 enables the comparison of Byte 0 with the current lower byte of the packet, Bit 1 enables comparison of Byte 1 and so on.

The receive data register clocks in the receive data coming from the MAC or the internal loopback path. It is 32 bits wide and samples rx_data each time rx_data_valid is asserted. To serialize the words for comparison with the pattern words a word counter controls a word wide multiplexer that puts words 0..1 of the received dword in two consecutive clock cycles to the compare logic. At Gigabit speed each 2 clock cycles a new dword arrives. At 100 Mbit and 10 Mbit speed each 8 clock cycles a new dword arrives. The word counter starts in the clock cycle after rx_data_valid is asserted with count 0b00 and stops after 2 clock cycles, if not restarted by another rx_data_valid.

The pattern pointer is used as read address at the read only port of the Pattern RAM. It is reset to zero at SOP and incremented by two each time the word counter is incremented.

The Pattern Match Statemachine controls the pattern match operation by initializing the match logic on SOP and enabling compare logic between SOP and EOP. It also controls the sampling of the receive data in the receive data register to avoid sampling of the receive status word.

The compare logic consists of 7 identical instances of the following components:

- A pattern length register holds the length of the pattern. It is initialized by SW.
- A pattern length counter is loaded at SOP from the pattern length register and is decremented by two each time the word counter is incremented. As soon as the pattern length counter for a pattern reaches zero, it signals the end of the pattern match operation. It can be read and written by SW for test purposes only.
- A word-wide comparator compares the received word to the current word of the pattern.
- A flip-flop signals matching data. It is set at SOP and reset as soon as the comparator signals a mismatch and the pattern mask bit for the current pattern byte is set. Also it is reset, if the byte enable for the current byte is not set, but the pattern mask bit is set. If the pattern mask bit for the current byte is not set, the byte is not compared, but treated as matching.
- A flip-flop signals the result of the pattern match operation. It is reset at SOP and samples the state of the matching data flag as soon as the complete pattern has been compared. If EOP is signaled, before the pattern length counter has expired, this flip-flop remains reset and the pattern does not match. It can be read out in the Status Register.

The Control/Status Register contains the following control bits:

- A radio button to enable/disable the Wake up Frame Logic. If disabled, all flip-flops, flags and counters of the Wake up Frame Logic are reset to zero. The pattern length registers and the other control bits can be set while the Wake up Frame Logic is disabled.
- A control bit to clear the result status bits for all patterns by SW.
- A control bit for each pattern to enable/disable comparison of incoming packets with the corresponding pattern.
- A status bit for each pattern to show the result of the last pattern match operation.

For writes first the lower three words of the 128 bit pattern word have to be written to the Pattern RAM Data Registers, then the upper 32 bits are written to the fourth Pattern RAM Data Register. In the clock cycle after the fourth Pattern RAM Data Register has been written, the complete 128 bit pattern word is written to the according Pattern RAM location.

On reads data is read directly out of the Pattern RAM. All four dwords are updated simultaneously.

2.12.2 Magic Packet frame detect

Magic Packet frame detect: The incoming data stream is searched for a so called “magic packet frame” that consists of 6 bytes of 0xFF followed by 16 iterations of the device’s MAC address (see chapter *3.3.2.57 Wake on LAN Control Registers* on page 203). If this sequence is found, a wake up event is created.

Note: Nevertheless the incoming packet must have a valid destination address or multicast address.

2.12.3 Link Change Monitoring

Link Change monitoring: Change of the link status from down to up causes a wake up event.

2.13 GMAC

The Gigabit Ethernet functionality is realized with an integrated GMAC module provided by Marvell in combination with the PHY module.

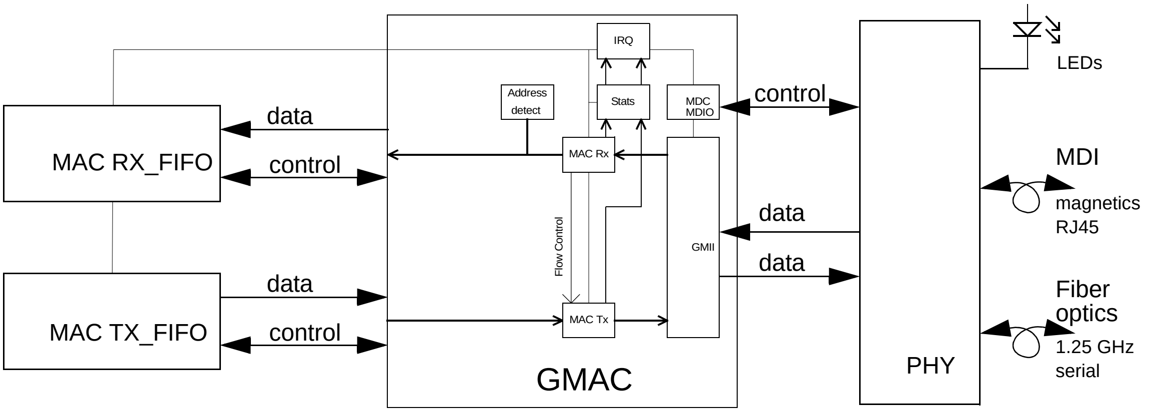
All registers of the GMAC are accessible by the CPU and are mapped into the I/O address space **GMAC Registers**.

The reset line /RESET of the GMAC is controlled by GMAC Reset.

The GMAC runs at core clock (clk_host, 125 MHz).

The interfaces to the rx and tx FIFOs have their own rx_clk and tx_clk.

Figure 14: GMAC and PHY integration



2.14 PHY

The PHY module is based on IP (Alaska 88E1111) provided by Marvell. (For detailed information, refer to "88E1111 Integrated 10/100/1000 Gigabit Ethernet Transceiver", Marvell, Doc. No. MV-S100649-00).

2.15 LEDs

The 88E8053 device supports 4 LED signals. All LEDs can be driven via register settings. The GPHY internal four LED lines can be driven in different modes (for detailed information, refer to "88E1111 Integrated 10/100/1000 Gigabit Ethernet Transceiver", Marvell, Doc. No. MV-S100707-00). List of all supported LED lines by the 88E8053 device:

LED_ACTn, LED_LINK10/100n, LED_LINK1000n, and LED_LINK

2.15.1 LED Capabilities of 88E8053

All PHY controlled LEDs may also be controlled directly by software (refer to "88E1111 Datasheet, Integrated 10/100/1000 Gigabit Ethernet Transceiver", Marvell, Register 24 and Register 25).

2.15.1.1 Speed LEDs

The LED_LINK10/100n, LED_LINK1000n, and LED_LINKn pins are controlled by PHY.

The LED_LINK10/100n pin is the parallel LED output for 100BASE-T link or speed. The LED_LINK10/100n pin indicates 100 Mbps link or speed if active.

The LED_LINK1000n pin is the output for 1000BASE-T link. The LED_LINKn pin is the parallel output for 10/100/1000BASE-T link.

2.15.1.2 Link LED

The LED_Link pin is controlled by the PHY.

(For detailed information refer to the 99E1111 Datasheet, Integrated 10/100/100 Gigabit Ethernet Transceiver, Marvell®, Register 24: LED Control and LED Interface).

2.16 VPD

The network adapter implements VPD as suggested by PCI Rev. 2.3. It is stored in a TWSI EEPROM (or in the SPI Flash Memory) and may be accessed through the **VPD Address Register** and **VPD Data Register**. These registers are mapped writable both in configuration and I/O address space.

For detailed description of the TWSI EEPROM see chapter *2.5 TWSI EEPROM* on page 22.

In absence of the TWSI EEPROM VPD is stored within the SPI Flash Memory. Then the VPD cannot be reprogrammed during operation of the device, because rewriting of the SPI Flash Memory is done by sector erase (32 kB).

2.17 TWSI Interface

The TWSI interface is controlled either by SW via the **Interface Register** or by HW via the **TWSI Control Register** and **TWSI Data Register**. If HW controlled TWSI accesses are used, the **Interface Register** must be set to inactive values (Clock = 1, Direction = 0, Data = 0).

The HW controlled Interface can be parameterized in several ways. The size of the target device of the TWSI access (and implicit the number of address bytes/bits to be used) and its devsel byte must be written together with the address to the **TWSI Control Register**. If the **TWSI Burst** bit is set, the TWSI Interface runs 4 byte bursts in page mode, assuming pages of 8 bytes. Invalid or erroneous HW controlled TWSI accesses that don't complete, can be stopped by writing a 1 to **TWSI Stop**. On completion of a HW controlled TWSI access an interrupt **IRQ TWSI Ready** is asserted.

2.18 Parity Generation/Check

PCI parity checking/generating follows PCI Specification as even parity on dwords.

2.18.1 Internal byte based parity checking/generating

Byte based parity is generated on data entering the PCI Transmit FIFOs and on data entering the MAC Receive FIFOs. For Internal Status words written to the RAM, parity is generated by the RAM interface.

Byte based parity is checked on data leaving the PCI Receive FIFOs and on data leaving the MAC Transmit FIFOs. Parity is also checked on data read from the RAM.

Each parity checker generates an Interrupt. All parity Interrupts are routed to the **Interrupt HW Error Source Register**.

NOTE: Even if an **Interrupt Parity Error** is generated, running operations are continued.

Parity on the PCI bus is generated, checked and reported following the PCI specification.

2.18.2 Parity Checking/Generating on PCI as target

Read data parity is generated for all read accesses to device resources (SPI Flash Memory, ASIC-Registers) in the ASIC.

Write data parity is checked for all write accesses to device resources (SPI Flash Memory, ASIC-Registers) in the ASIC.

Address parity is checked for all address phases running on the bus.

If a write data parity error is detected, **Parity Error** is set. Status register bit (Offset: 0x06, page 60) **PERR#** is asserted, if **Parity Report Response Enable** is set.

If an address parity error is detected, **Parity Error** is set. Status register bit (Offset: 0x06, page 60) **SERR#** is asserted and **Signaled Error** is set, if **SERR# enable** and **Parity Report Response Enable** are set.

2.18.3 Parity Checking/generating on PCI as master

Write data parity is generated for all write accesses to system memory.

Read data parity is checked for all read accesses from system memory.

Address parity is generated for all address phases generated on the bus.

If a read data parity error is detected, **Parity Error** is set. Status register bit (Offset: 0x06, page 60) **PERR#** is asserted and **Data Parity Error detected** is set, if **Parity Report Response Enable** is set.

If **PERR#** is sampled asserted on a write access, **Parity Error** is set. **Data Parity Error detected** is set, if **Parity Report Response Enable** is set.

If **Data Parity Error detected** is set, interrupt **IRQ Master Error** is set.

If **Parity Error** is set, interrupt **IRQ Status** is set (see also **Interrupt Register**).

Section 3. Register Description

3.1 Legend

Throughout this document frame and packet (also STF and STP) are used synonymously.

Dword stands for double word (4 bytes).

Qword stands for quadruple word (8 bytes).

Register descriptions

The following conventions are valid for register descriptions:

- Write:
  - ne = no effect (read only register)
  - yes = writable
  - sh = special handling as described
  - exec = execution of this command, if appropriate bit is set
  - ITO = writable during initialization and for tests only
  - TO = writable for tests only
- Read:
  - aw = as written
  - value = as defined by itself
  - given number (fixed values typically)

Commands (single bit) in Control Registers:

The following conventions are valid for commands in control registers:

- Commands are executed, if appropriate bit is set
- Read value as defined.

Exclusive commands (xxx Start/Stop, xxx On/Off):

The following conventions are valid for exclusive commands:

- Commands are executed, if appropriate bit is set to 1.
- Setting both commands to 1, has no effect.
- Status is readable: 0b01 or 0b10.

- Reset Value:
  - <blank> = fixed value or value directly from input pin
  - <value> = reset to <value> only by Power on and **HW Reset**
  - <value> (HW) = reset to <value> only by Power on and **HW Reset**
  - <value> (SW) = reset to <value> by Power on, **HW Reset** and **SW Reset**

Reserved registers are still empty within the address space.

Reserved (legacy) registers are not used within this applications but have to be left empty due to former SW compatibility.

3.2 PCI-Express Configuration Register File

The Configuration Register File is mapped into the Control Register File block 7 (only first 128 Bytes) and entirely at blocks 56 up to 60.

3.2.1 Overview and Address Map

The table below depicts the layout of the configuration space.

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address
Header Region				
Device ID		Vendor ID		0x00
Status		Command		0x04
Class Code			Revision ID	0x08
BIST	Header Type	Latency Timer	Cache Line Size	0x0c
Base Address (1st)/Lower				0x10
Base Address (1st)/Upper				0x14
Base Address (2nd)				0x18
Reserved (Unused Base Address)				0x1c
Reserved (Unused Base Address)				0x20
Reserved (Unused Base Address)				0x24
Reserved				0x28
Subsystem ID		Subsystem Vendor ID		0x2c
Expansion Rom Base Address				0x30
Reserved			New Cap Ptr	0x34
Reserved				0x38
Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	0x3c

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address
Device Dependent Region				
Our Register 1				0x40
Our Register 2				0x44
PM Capabilities		Next Item Ptr	PM Cap ID	0x48
PM Data Reg	Reserved	PM Control/Status		0x4c
VPD Address Register		Next Item Ptr	VPD Cap ID	0x50
VPD Data Register				0x54
TWSI EEPROM Loader Control Register		Reserved		0x58
MSI Message Control		Next Item Ptr	MSI Cap ID	0x5c
MSI Message Address (Lower)				0x60
MSI Message Address (Upper)				0x64
Reserved		MSI Message Data		0x68
PCI-X Command Register		Reserved	PCI-X Cap	0x6c
PE Status				0x70
Calibration Status Register		Calibration Control Register		0x74
Reserved	Retry Counter	Discard Counter		0x78
Our Status Register				0x7c
Reserved				0x80:dc
PCI Express Capability				
PE Capabilities Register		Next Item Ptr	PE Cap ID	0xe0
Device Capabilities				0xe4
Device Status		Device Control		0xe8
Link Capabilities				0xec
Link Status		Link Control		0xf0
Reserved				0xf4:0xfc

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address
PCI Express Extended Capabilities				
Advanced Error Reporting Enhanced Capability Header				0x0100
Uncorrectable Error Status Register				0x0104
Uncorrectable Error Mask Register				0x0108
Uncorrectable Error Severity Register				0x010c
Correctable Error Status Register				0x0110
Correctable Error Mask Register				0x0114
Advanced Error Capabilities and Control Register				0x0118
Header Log Register				0x011c
				0x0120
				0x0124
				0x0128
Reserved				0x012c:0x01fc
Transaction Layer Control Register				0x0200
Transaction Layer Status Register				0x0204
Data link Layer Control Register				0x0208
Data Link Layer Status Register				0x020c
PE Physical Layer Control Register				0x0210
PE Physical Layer Status Register				0x0214
Reserved				0x0218:0x021c
PE Completion Timeout Register				0x0220
PE Flow Control Register				0x0224
PE Ack Timer for 1x Link				0x0228
PE Ack Timer for 4x Link				0x022c
Reserved				0x0230:0x0ffc

3.2.2 Registers of PCI Header Region

3.2.2.1 Vendor ID Register

Address: 0x00 Width [bit]: 16

The PCI SIG has allocated 0x1148 to Marvell® as a unique identifier.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Vendor ID Register					
15:0		Identifies Marvell as manufacturer of the PCI device.	ne	0x11ab	0x11ab

3.2.2.2 Device ID Register

Address: 0x02 Width [bit]: 16

The Device ID register is a 16-bit register that uniquely identifies the PCI device within the Marvell® product line.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Device ID Register					
15:0		Identifies the PCI device within Marvell's product line.	ne	0x4360	0x4360

3.2.3 Registers of Header Region

3.2.3.1 Command Register

Address: 0x04 Width [bit]: 16

All bits are reloadable out of the SPI Flash Memory or TWSI EEPROM, except for fixed value bits.

Bit	Name	Description	Write	Read	Reset value
Command Register					
15:11	**Reserved**				
10	**INTDIS**	Disables the device from asserting INTx#. 1 = Disables the assertion of its INTx# signal. 0 = Enables the assertion of its INTx# signal. Refer to MSI Enable bit. If MSI Enable bit is set to 1, device is prohibited from using INTx#.	yes	aw	0
9	**Reserved**		ne	0	0
8	**SERREN**	1= Enables reporting of non-fatal and fatal errors to the Root Complex. Note that errors are reported if enabled either through this bit or through the PCI-Express specific bits in the Device Control Register	yes	aw	0
7	**Reserved**	<PCI_Express>Not applied to PCI-Express. Fixed value 0.	ne	0	0
6	**PERREN**	Parity Error Report Response Enable. 1 = Parity error reporting is enabled 0 = Parity error reporting is disabled.	yes	aw	0
5:3	**Reserved**		ne	0	0
2	**BMEN**	Bus Master enable Controls the ability of a PCI Express agent to issue memory and I/O read/write requests. 1= Bus master accesses are enabled, 0 =Bus master accesses are disabled.	yes	aw	0
1:0	**Reserved**				

3.2.3.2 Status Register

Address: 0x06 Width [bit]: 16

All bits are reloadable out of the SPI Flash Memory or TWSI EEPROM, except for fixed value bits.

Bit	Name	Description	Write	Read	Reset value
Status Register					
15	**PERR**	Parity Error. 1: This bit is set by a device whenever it receives a poisoned TLP, regardless of the state of the Parity Error Enable bit.	sh	value	0
14	**SERR**	Signaled **SERR#**. 1: This bit is set when a device sends an ERR_FATAL or ERR_NONFATAL message, and the SERREN bit in the Command Register is 1.	sh	value	0
13	**RMABORT**	Received Master Abort. 1: This bit is set when a Requestor receives a Completion with Unsupported Request Completion Status.	sh	value	0
12	**RTABORT**	Received Target Abort. 1: This bit is set when a Requestor receives a Completion with Completer Abort Completion Status.	sh	value	0
11:9	**Reserved**				
8	**DATAPERR**	Data Parity Error detected. 1: This bit is set by a Requestor if its Parity Error Enable bit is set when either of the following two conditions occur: - Requestor receives a Completion marked poisoned - Requestor poisons a write Request.	sh	value	0
7:5	**Reserved**		ne	0	0
4	**NEWCAP**	New capabilities bit 1: New capabilities list implemented 0: New capabilities list not implemented	ne	1	1
3	**INTSTA**	1: Indicates that an INTx interrupt message is pending internally to the device.	ne	value	0
2:0	**Reserved**				

3.2.3.3 Revision ID Register

Address: 0x08 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM

Bit	Name	Description	Write	Read	Reset value
	Revision ID Register				
7:0		Specifies the PCI device revision number/ Rev. 0.0.	ne	0x00	0x00

3.2.3.4 Class Code Register

Address: 0x09 Width [bit]: 3 x 8

The Class Code Register is used to identify the generic function of the PCI device. The register consists of three byte-size fields.

The Subclass Register is reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
	Programming Interface Register, Lower Byte				
7:0		Specifies the programming interface. Fixed Value = 0	ne	0	0

Bit	Name	Description	Write	Read	Reset value
	Sub-Class Register, Middle Byte				
7:0		Identifies the network controller as an “Ethernet Controller”.	ne	0x00	0x00

Bit	Name	Description	Write	Read	Reset value
	Base-Class Register, Upper Byte				
7:0		Broadly classifies the function of the PCI device as network controller. Fixed Value = 0x02	ne	0x02	0x02

3.2.3.5 Cache Line Register

Address: 0x0c Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM (not recommended).

Bit	Name	Description	Write	Read	Reset value
Cache Line Size Register					
7:0		Reserved	0	0	0

3.2.3.6 Latency Timer Register

Address: 0x0d Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM (not recommended).

Bit	Name	Description	Write	Read	Reset value
Latency Timer Register					
7:0		Reserved			

3.2.3.7 Header Type Register

Address: 0x0e Width [bit]: 8

Bit	Name	Description	Write	Read	Reset value
Base-Class Register					
7:0		Reserved			

3.2.3.8 Built-in Self Test Register (BIST)

Address: 0x0f Width [bit]: 8

Bit	Name	Description	Write	Read	Reset value
Built-in Self Test Register					
7:0		BIST is not supported . Fixed value = 0.	ne	0	

3.2.3.9 Base Address Register (1st)

Address: 0x010 Width [bit]: 2 x 32

The 1st Base Address Register uses two 32-bit registers that determine the location of the PCI device in memory space, if memory mapping is enabled.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Base Address Register (1st)/Lower (Address: 0x010, Width [bit]: 32)					
31:14	**Lower MEM-BASE Address**	18 bits of lower mem base address.	yes	aw	0
13:4	**MEMSIZE**	Memory size requirements. Fixed value 0: Memory space requirement of 16384 bytes.	ne	0	
3	**PREFEN**	Prefetch enable. Fixed value 0: Prefetching is not allowed. (Memory Write Byte Merging is not tolerable).	ne	0	
2:1	**Memory Type**	Memory type. 0b00: Base register is 32 bits wide, and mapping can be done anywhere in the 32-bit memory space. 0b10: Base register is 64 bits wide and can be mapped anywhere in the 64-bit address space. **Memory Type** may be reloaded out of the SPI Flash Memory or TWSI EEPROM (further memory types).	ne	0x02	0x02
0	**MEMSPACE**	Memory space indicator. Fixed value = 0: This **Base Address Register** describes a memory base address.	ne	0	
Base Address Register (1st)/Upper (Address: 0x014, Width [bit]: 32)					
31:0	**Upper MEM-BASE**	Upper 32 bits of memory base address.	yes	aw	0

3.2.3.10 Base Address Register (2nd)

Address: 0x018 Width [bit]: 32

The 2nd Base Address Register determines the location of the PCI device in the I/O space.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

If **En IO Mapping** (in **Our Register 1,** bit 23) is disabled, this location is treated like **reserved** locations.

Bit	Name	Description	Write	Read	Reset value
Base Address Register (2nd)					
31:8	**IOBASE**	I/O base address most significant 24 bits.	yes	aw	0
7:2	**IOSIZE**	I/O size requirements. Fixed value 0x0: I/O space requirement of 256 bytes.	ne	0	
1	**Reserved**				
0	**IOSPACE**	I/O space indicator. 1: This **Base Address Register** describes an I/O base address.	ne	1	1

3.2.3.11 Subsystem Vendor ID Register

Address: 0x02c Width [bit]: 16

The Subsystem Vendor ID register may be used for customizing OEM versions. The subsystem Vendor ID is allocated by the PCI SIG.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Subsystem Vendor ID Register					
15:0	**Subsystem Vendor ID**	Identifies the subsystem vendor. Must be a valid non-zero value.	ne	0x11ab	0x11ab

3.2.3.12 Subsystem ID Register

Address: 0x02e Width [bit]: 16

The Subsystem ID register may be used for customizing OEM versions.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Subsystem ID Register					
15:0	**Subsystem ID**	Identifies the subsystem. Must be a valid non-zero value. (Default value: Device ID)	ne	aw	0x4360

Should be reloaded from SPI Flash Memory or TWSI EEPROM with the Subsystem ID of the related manufacturing option.

3.2.3.13 Expansion Rom Base Address Register

Address: 0x030 Width [bit]: 32

The Expansion Rom Base Address Register is a 32-bit register that determines the base address and size information of the Expansion Rom.
Within the SPI Flash Memory 96 kB may be used for boot code.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

If **En Eprom** (in **Our Register 1**, bit 22) is disabled, this location is treated like **reserved** locations.

Bit	Name	Description	Write	Read	Reset value
Expansion Rom Base Address Register					
31:17	**Rombase**	ROM base address significant 15 bits.	yes	aw	0
16:14	**Rombase/size**	Treated as **Rombase** or **Romsize** depending on settings of **Pagesize** (**Our Register 1**, bits 21:20).	yes ne	aw 0	0
13:11	**Romsize**	ROM size requirements. Fixed value 0: Memory space requirement of 16 kB or higher.	ne	0	

Bit	Name	Description	Write	Read	Reset value
10:1	**Reserved**				
0	**ROMEN**	Address decode enable. Read/write accessible. 0: The device's Expansion ROM address space is disabled. 1: And **MEMEN** = 1 (**Command Register**, bit 1), the device's Expansion ROM address space is enabled.	yes	aw	0

3.2.3.14 New Capabilities Pointer (New Cap Ptr)

Address: 0x034 Width [bit]: 8

The New Capabilities Pointer Register is a 8-bit register pointing to the first element in the New Capabilities List.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
New Capabilities Pointer Register					
7:0	**New Capabilities Pointer**	Points to the New Capabilities List	ne	0x48	0x48

3.2.3.15 Interrupt Line Register

Address: 0x03c Width [bit]: 8

Bit	Name	Description	Write	Read	Reset value
Interrupt Line Register					
7:0		The Interrupt Line Register is used to communicate interrupt line routing information. POST software writes the routing information into this register as it initializes and configures the system. The value of this register indicates to which input of the system interrupt controller(s) the device's interrupt pin is connected. Device drivers and operating systems may use this information to determine priority and vector information. The Interrupt Line Register is not modified by the PCI device. It has no effect on the operation of the device.	yes	aw	0

3.2.3.16 Interrupt Pin Register

Address: 0x03d Width [bit]: 8

Bit	Name	Description	Write	Read	Reset value
Interrupt Pin Register					
7:0		Fixed value 0x01:The PCI device uses the interrupt pin **INTA#**.	ne	0x01	

3.2.3.17 Min_Gnt Register

Address: 0x03e Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Min_Gnt Register					
7:0		This read-only register specifies the PCI device's desired settings for Latency Timer value. The value specifies - in units of 1/4 microseconds - the burst period needed by the PCI device assuming a clock rate of 33 MHz. (64 qwords x 30 ns) = 1.92 µs 1.92 µs/0.25 µs = 8	ne	0x08	0x08

3.2.3.18 Max_Lat Register

Address: 0x03f Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Max_Lat Register					
7:0		**Reserved**	ne	0x00	0x00

3.2.4 Registers of Device Dependent Region

3.2.4.1 Our Register 1 and Our Register 2

Address: 0x040 Width [bit]: 2 x 32

These are the first two used locations in the “device dependent region”. The default values are chosen for the most common environments.

Modifications may be handled as manufacturing option, driver options, or dedicated configuration software.

Most of the switches in **Our Register 1** and **2** are not intended for use at run time. Manufactured PCI devices may come up with different settings than defined as **Reset value** (if reloaded out of the SPI Flash Memory or TWSI EEPROM).

Reloadable out of the SPI Flash Memory or TWSI EEPROM. The fields marked in column **write** with “ne” are writable only in testmode. The fields marked with “yes” are writable in configuration space and with normal accesses to the **Control Register File**

.

Bit	Name	Description	Write	Read	Reset value
Our Register 1 (Address: 0x40, Width [bit]: 32)					
31	**Run_PiG**	1: Enable Plug In Go.	ne	1	1
30	**DLL_dis**	PCI DLL disable reset strapping (disable = 1). It is a debug bit (if everything is ok, should be always set to ‘1’)	ne	1	1
29	**Reserved**				
28	**PHY Coma Mode**	Set PHY to Coma Mode 1: Coma mode 0: Normal operation Lowest possible Power Mode, no core clock	ne	aw	0
27	**Reserved**				
26	**PHY Power Down Mode**	Set PHY to Power Down Mode 1: Power Down mode 0: Normal operation PHY Power Down Mode (low Power mode, only core clock active).	yes	aw	0
25	**Reserved**				
24	**En Boot**	Boot enable, for software purposes only 1: Don’t boot with expansion ROM code. 0: Boot with expansion ROM code	yes	aw	0
23	**En IO Mapping**	Controls mapping of the **Control Register File** to the I/O space (manufacturing option). 1: Address decoding for I/O accesses enabled. 0: Any address decoding for I/O accesses is disabled (see also **Base Address Register (2nd)**).	ne	1	1

Bit	Name	Description	Write	Read	Reset value
22	**En Eprom**	Controls mapping of the SPI Flash Memory to the memory space. 1: Address decoding for memory accesses enabled. 0: Any address decoding for memory accesses is disabled (see also **ROM Base Address Register**).	ne	0	0
21:20	**Pagesize<1:0>**	Pagesize/SPI Flash Memory Defines the size, which is mapped to the system (see **Romsize/Rombase**). 0x3: 128 kB 0x2: 64 kB 0x1: 32 kB 0x0: 16 kB	ne	0x3	0x3
19	**Reserved**				
18:16	**Page Reg<2:0>**	Page Register Selects the page of the SPI Flash Memory space, which is mapped to the system. 0x0: Page 0, 0x1: Page 1, ... 0x7: Page 7	yes	aw	0
15	**PEX_LegNat_Sel**	PEX: 1: Device in PEX legacy mode; 0: Device in PEX native mode. Reset by Power on reset.	yes	aw	0
14:0	**Reserved**		yes	aw	

Bit	Name	Description	Write	Read	Reset value
Our Register 2 (Address: 0x44, Width [bit]: 32)					
31:24	**VPD Write Thr**	Defines the first address of the writable VPD area in steps of 128 Bytes. Default value is 128 Bytes = address 0x80. Last address of writable VPD area is 255. Higher addresses belong to the configuration data also stored within TWSI EEPROM.	ne	0x01	0x01
23:17	**VPD Devsel**	Defines the Device Select Byte for the TWSI EEPROM used for VPD storage. Default value is 0b1010000. Hint: **VPD Devsel must not be overwritten via TWSI EEPROM**. This may lead to a complete damage of the board (TWSI EEPROM must be changed afterwards!!!)	ne	0x50	0x50
16:14	**VPD ROM Size**	Defines the size of the assembled TWSI EEPROM in Bytes. 0x0: 256 Bytes 0x1: 512 Bytes 0x2: 1024 Bytes 0x3: 2048 Bytes 0x4: 4096 Bytes 0x5: 8192 Bytes 0x6: 16384 Bytes 0x7: 32768 Bytes Default value is 2048 Bytes. If any other size is used, this field must be reprogrammed out of the SPI Flash Memory. Due to currently used addressing procedure via TWSI bus only applications up to size 2048 Bytes are supported.	ne	0x3	0x3
13:0	**Reserved**				

3.2.4.2 Power Management Capability ID Register (PM Cap ID)

Address: 0x048 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Power Management Capability ID Register					
7:0	**Cap ID**	Power Management Capabilities ID	ne	0x01	0x01

3.2.4.3 Power Management Next Item Pointer

Address: 0x049 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Power Management Next Item Pointer					
7:0	**Next Item Ptr**	Pointer to the next item in the capabilities list	ne	0x50	0x50

3.2.4.4 Power Management Capabilities Register

Address: 0x4a Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Only the bits are mentioned which differ from the PCI Configuration File.

Bit	Name	Description	Write	Read	Reset value
Power Management Capabilities Register					
15:11	**PME Support**	Power Management Event Support: Specifies the power state in which the signal PME# may be asserted. If no Vaux is available, bit 15 is forced to zero signaling no PME# support in $D3_{cold}$	ne		

Bit	Name	Description	Write	Read	Reset value
15		1: PME# can be asserted from $D3_{cold}$, if Vaux is available 0: PME# cannot be asserted from $D3_{cold}$, if Vaux is not available		1 (0 if no Vaux)	1 (0 if no Vaux)
14		1: PME# can be asserted from $D3_{hot}$		1	1
13		1: PME# can be asserted from D2			
12		1: PME# can be asserted from D1		1	1
11		1: PME# can be asserted from D0		1	1
10	**D2 Support**	D2 Support 1: The PCI device supports D2 Power Management State.	ne		
9	**D1 Support**	D1 Support 1: The PCI device supports D1 Power Management State.	ne	1	1
8:6	**Reserved**	Reserved, but reloadable out of the SPI Flash Memory or TWSI EEPROM for changes in the PCI Specification.	ne	0b000	0b000
5	**DSI**	Device Specific Initialization: 1: The PCI device requires device specific initialization 0: The PCI device does not require device specific initialization	ne	0	0
4:3	**Reserved**				
2:0	**Version**	The PCI device complies with Revision 1.1 of the PCI Power Management Interface Specification	ne	0x2	0x2

3.2.4.5 Power Management Control/Status Register

Address: 0x04c Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Power Management Control/Status Register					
15	**PME Status**	1: Indicates that PME# has been asserted by the PCI device. Reset by Power on reset and when written with 1.	sh	value	0
14:13	**Data Scale**	Indicates the scaling factor to be used when interpreting the value of the **Data Register**. The read value depends on the setting of the **Data Select** field.	ne	value	0b01
12:9	**Data Select**	This 4-bit field is used to select which data is to be reported through the Data Register and **Data Scale** field.	yes	aw	0
8	**PME En**	1: Enables PME# generation. Reset by Power on reset	yes	aw	0
7:2	**Reserved**				
1:0	**Power State**	Controls the Power Management State of the PCI device. The PCI device supports all power management states.	yes	aw	0

The 16 2-bit **Data Scale** fields and the 16 8-bit **Data Register** values that can be selected by the **Data Select** field, are reloadable out of the SPI Flash Memory or TWSI EEPROM by writing complete 32-bit wide sets of (**Data Select**, **Data Scale**, **Data**) to the **Power Management Control/Status** and **Data Register**.

Warning: To modify the contents of any **Data Scale** or **Data** field the **Data Scale** field **MUST** always be written with the desired **Data Select** value in the same 32-bit access!

3.2.4.6 Power Management Data Register

Address: 0x04f Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
	Power Management Data Register				
7:0	**Data**	This read-only register is used to report the state dependent data requested by the **Data Select** field. The value of this register is scaled by the value reported by the **Data Scale** field.	ne	value	see Power Man-age-ment Data Table

3.2.4.7 Power Management Data Table

Data and **Data Scale** are hidden registers accessible through the **Power Management Control/Status Register (Data Select).**

Data Scale is writable by the EPROM Loader by writing to **Power Management Control/Status Register.**

Data is writable by the EPROM Loader by writing to **Power Management Data Register.**

Data and **Data Scale** are reloaded from the SPI Flash Memory or TWSI EEPROM with values matching the manufacturing option. See “SPI Flash Memory Reloads” in the PCI device’s application specification”.

Value in Data Select	**Meaning**	**Data (8 bit) Reset Value**	**Data Scale (2 bit) Reset Value**	**[Watt] with Reset Values**
0	D0 Power consumed	0x13	0b01	1.9
1	D1 Power consumed	0x0c	0b01	1.2
2	D2 Power consumed	0x0c	0b01	1.2
3	D3 Power consumed	0x0c	0b01	1.2
4	D0 Power dissipated	0x13	0b01	1.9
5	D1 Power dissipated	0x0c	0b01	1.2
6	D2 Power dissipated	0x0c	0b01	1.2
7	D3 Power dissipated	0x0c	0b01	1.2
8:15	For multifunction devices only	0	0	0

3.2.4.8 VPD Capability ID Register (VPD Cap ID)

Address: 0x050 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	**Name**	**Description**	**Write**	**Read**	**Reset value**
VPD Capability ID Register					
7:0	**Cap ID**	VPD Capabilities ID	ne	0x03	0x03

3.2.4.9 VPD Next Item Pointer

Address: 0x051 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	**Name**	**Description**	**Write**	**Read**	**Reset value**
VPD Next Item Pointer					
7:0	**Next Item Ptr**	Pointer to the next item in the capabilities list.	ne	0x5c	0x5c

3.2.4.10 VPD Address Register

Address: 0x052 Width [bit]: 16

The **VPD Address Register** and the **VPD Data Register** control a TWSI interface, which runs a 100 kHz protocol to an external TWSI EEPROM.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Also writable in I/O space.

Bit	Name	Description	Write	Read	Reset value
VPD Address Register					
15	**Flag**	Starts the VPD data transfers, determines its direction and signals its completion by being toggled. If written 1, a VPD write is started. Set to 0 after completion. If written 0, a VPD read is started. Set to 1 after completion.	exec	value	0
14:0	**VPD Address**	Address of the VPD contents to be written / read.	yes	aw	0x00

3.2.4.11 VPD Data Register

Address: 0x054 Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Also writable in I/O space.

Bit	Name	Description	Write	Read	Reset value
VPD Data Register					
31:0	**VPD Data**	Must be written before VPD Address Register for VPD write. Contains VPD read Data after completion of VPD read.	yes	aw	0x00

3.2.4.12 TWSI EEPROM Control Register

Address: 0x05a Width [bit]: 16

The **TWSI EEPROM Control Register** controls the TWSI EEPROM Loader.

Reloadable out of the SPI Flash Memory and can be written in test mode.

Bit	Name	Description	Write	Read	Reset value
TWSI EEPROM Control Register					
15	**Flag**	Starts and stops the data transfer. If written 1, the TWSI EEPROM Loader is started. If written 0, the TWSI EEPROM Loader is stopped.	ne	0	0
14:0	**TWSI EEPROM Address**	Start address for TWSI EEPROM Loader. Should be minimum 256 (0x100) and in 8 byte steps.	ne	0x100	0x100

3.2.4.13 MSI Capability ID Register (MSI Cap ID)

Address: 0x05c Width [bit]: 8

The device is capable of Message Signaled Interrupt (MSI) handling.

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
MSI Capability ID Register					
7:0	**Cap ID**	MSI Capabilities ID	ne	0x05	0x05

3.2.4.14 MSI Next Item Pointer

Address: 0x05d Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
MSI Next Item Pointer					
7:0	**Next Item Ptr**	Pointer to the next item in the capabilities list.	ne	0x6c	PEX: 0x0e

3.2.4.15 MSI Message Control

Address: 0x05e Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
MSI Message Control					
15:8	**Reserved**		ne	0	0
7	**64 Bit Addr capable**	1: This device is capable of generating a 64-bit message address 0: The device is not capable of generating a 64-bit message address	ne	1	1
6:4	**Multiple Message Enable**	Defines the number of allocated messages 0b000: 1 The implementation supports one allocated Message.	yes	value	0
3:1	**Multiple Message Capable**	System software reads this field to determine the number of requested messages. 0b000: 1 There is one requested message.			
0	**MSI Enable**	1: MSI is used to request service. INTA# is disabled. 0: INTA# is used to request service MSI is disabled			

3.2.4.16 MSI Message Address

Address: 0x060 Width [bit]: 2 x 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
MSI Message Lower Address (Address: 0x060, Width [bit]: 32)					
31:2		System-specified message address If **MSI Enable** is set, the contents of this register specify the DWORD aligned address for the MSI memory write transaction.	yes	value	0x00
1:0	**Reserved**				
MSI Message Upper Address (Address: 0x064, Width [bit]: 32)					
31:0		System-specified message upper address If **MSI Enable** is set, the contents of this register (if non-zero) specify the upper 32-bits of a 64-bit message address If the contents of this register are zero, the device uses the 32 bit address specified by the **MSI Message Lower Address**.	yes	value	0x00

3.2.4.17 MSI Message Data

Address: 0x068 Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
MSI Message Data					
31:16	**Reserved**				
15:0	**Message Data**	System-specified message. If **MSI Enable** is set, the **Message Data** is driven onto the lower word of the memory write transaction's data phase. The **Multiple Message Enable** field (bits 6:4 of the **Message Control Register**) defines the number (only one message supported by the chip)	yes	value	0x00

3.2.4.18 PCI Express Status

Address: 0x070 Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM

Bit	Name	Description	Write	Read	Reset value
PCI Express Status					
31:30	**Reserved**				
29:16		<PCI_Express>Not applied to PCI-Express. Fixed value 0.	ne	0	0
15:8	**Request ID (Bus Number)**	Request ID is the combination of a Requester's Bus Number, Device Number, and Function Number that uniquely identifies the Requester. The Bus Number is updated with each Configuration Write transaction.	ne	value	0
7:3	**Request ID (Device Number)**	Request ID is the combination of a Requester's Bus Number, Device Number, and Function Number that uniquely identifies the Requester. The Device Number is updated with each Configuration Write transaction.	ne	value	0
2:0	**Request ID (Function Number)**	The Function Number is part of the Requester ID. Fixed value: 0b000	ne	0x00	0x00

3.2.4.19 Calibration Control/Status Register

Address: 0x074, 0x076Width [bit]:16

<PCI_Express>Not applied to PCI-Express. Fixed value 0.

3.2.4.20 Discard Counter, Retry Counter

Address: 0x078, 0x07aWidth [bit]:16, 8

<PCI_Express>Not applied to PCI-Express. Fixed value 0.

3.2.4.21 Our Status Register

Address: 0x07c Width [bit]: 32

Bit	Name	Description	Write	Read	Reset value
Our Status Register					
31	**Reserved**		ne	value	
30	**Reserved**		ne	value	
29:28	**Reserved**		ne	value	
27	**Reserved**		ne	value	
26	**Reserved**		ne	value	0
25:24	**DLL Err**	DLL status indication: 0x0:No error 0x1:Delay line pointer at start, and down count 0x2:Delay line pointer at end, and up count 0x3:Reserved Read only register	ne	value	
23:20	**DLL Row**	DLL row counters values. Calculate the TAPs number, using row and column numbers. Read only register	ne	value	
19:16	**DLL Col**	DLL column counters values. Calculate the TAPs number, using row and column numbers. Read only register	ne	value	
15:8	**Reserved**				
7:0	**Reserved**				

3.2.4.22 PE Capability ID Register (PM Cap ID)

Address: 0x0e0 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Capability ID Register					
7:0	**Cap ID**	PCI Express Capabilities ID	ne	0x10	0x10

3.2.4.23 PE Next Item Pointer

Address: 0x0e1 Width [bit]: 8

Reloadable out of the SPI Flash Memory or TWSI EEPROM

Bit	Name	Description	Write	Read	Reset value
PE Next Item Pointer					
7:0	**Next Item Ptr**	Pointer to the next PCI capability structure.	ne	0	0

3.2.4.24 PE Capabilities Register

Address: 0x0e2 Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Capabilities Register					
15:14	**Reserved**				
13:9	**Interrupt Message Number**	If this function is allocated more than one MSI interrupt number, this register is required to contain the offset between the base Message Data and the MSI Message that is generated when any of the status bits in either the Slot Status register or the Root Port Status register of this capability structure are set. Hardware is required to update this field so that it is correct if the number of MSI Messages assigned to the device changes.	ne	0	0
8	**Reserved**				

Bit	Name	Description	Write	Read	Reset value
7:4	**Device/Port Type**	Indicates the type of PCI Express logical device. Defined encodings are: 0b0000: PCI Express Endpoint device 0b0001: Legacy PCI Express Endpoint device 0b0100: Root Port of PCI Express Root Complex 0b0101: Upstream Port of PCI Express Switch 0b0110: Downstream Port of PCI Express Switch 0b0111: PCI Express-to-PCI/PCI-X Bridge 0b1000: PCI/PCI-X to PCI Express Bridge.	ne	1	1
3:0	**Capability Version**	PCI Express capability structure version number.	ne	0x01	0x01

3.2.4.25 Device Capabilities Register

Address: 0x0e4 Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Device Capabilities Register					
31:28	**Reserved**				
27:26	**Captured Slot Power Limit Scale**	It specifies the scale used for the Slot Power Limit Value. Range of Values: 0b00: 1.0x 0b01: 0.1x 0b10: 0.01x 0b11: 0.001x	ne	0	0
25:18	**Captured Slot Power Limit Value**	Specifies the upper limit on power supplied by slot in combination with the Slot Power Limit Scale value. Power limit (in Watts) calculated by multiplying the value of this field by the value in the Slot Power Limit Scale field.	ne	0	0
17:15	**Reserved**				

Bit	Name	Description	Write	Read	Reset value
14	**Power Indicator Present**	1: indicates that a Power Indicator is implemented on the PCI device.	ne	0	0
13	**Attention Indicator Present**	1: indicates that an Attention Indicator is implemented on the PCI device.	ne	0	0
12	**Attention Button Present**	1: indicates that an Attention Button is implemented on the PCI device.	ne	0	0
11:9	**Endpoint L1 Acceptable Latency**	This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. Power management software uses the reported L1 Acceptable Latency number for comparison with the L1 Exit Latencies reported (see below) by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether Active State Link PM L1 entry can be used with no loss of performance. Defined encodings are: 0b000: Less than 1 µs 0b001: 1 µs to less than 2 µs 0b010: 2 µs to less than 4 µs 0b011: 4 µs to less than 8 µs 0b100: 8 µs to less than 16 µs 0b101: 16 µs to less than 32 µs 0b110: 32 µs up to 64 µs 0b111: More than 64 µs	ne	0b111	0b111

Bit	Name	Description	Write	Read	Reset value
8:6	**Endpoint L0s Acceptable Latency**	This field indicates the acceptable total latency that an Endpoint can withstand due to the transition from L0s state to the L0 state. Power management software uses the reported L0s Acceptable Latency number for comparison with the L0s exit latencies reported by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether Active State Link PM L0s entry can be used with no loss of performance. Defined encodings are: 0b000: Less than 64 ns 0b001: 64 ns to less than 128 ns 0b010: 128 ns to less than 256 ns 0b011: 256 ns to less than 512 ns 0b100: 512 ns to less than 1 µs 0b101: 1 µs to less than 2 µs 0b110: 2 µs up to 4 µs 0b111: More than 4 µs	ne	0b111	0b111
5	**Extended Tag Field Supported**	This field indicates the maximum supported size of the Tag field. 1: 8-bit Tag field supported 0: 5-bit Tag field supported	ne	0	0
4:3	**Phantom Functions Supported**	This field indicates the support of using unclaimed function numbers to extend the number of outstanding transactions allowed by logically combining unclaimed function numbers (called Phantom Functions) with the Tag identifier. Note that Phantom Function support for the Device must be enabled by the corresponding control field in the Device Control register. Not supported according to bit 9 of Device Control Register.	ne	0	0
2:0	**Max_Payload_Size Supported**	This field indicates the maximum payload size in bytes that the device can support for TLPs. Defined encodings are: 0b000: 128 B maximum payload size	ne	0	0

3.2.4.26 Device Control Register

Address: 0x0e8 Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Device Control Register					
15	**Reserved**				
14:12	**Max_Read_Request_Size**	This field sets the maximum Read Request size for the Device as a Requester in bytes. The Device must not generate read requests with size exceeding the set value. Defined encodings for this field are: 0b000: 128 B max read request size 0b001: 256 B max read request size 0b010: 512 B max read request size 0b011: 1024 B max read request size 0b100: 2048 B max read request size 0b101: 4096 B max read request size 0b110: Reserved 0b111: Reserved	yes	aw	0x02
11	**Enable No Snoop**	1: The device is permitted to set the No Snoop bit in the Requester Attributes of initiated transactions that do not require hardware enforced cache coherency. Fixed value 0, not supported.	ne	0	0
10	**Auxiliary (AUX) Power PM Enable**	1: enables a device to draw AUX power independent of PME AUX power. Reset by Power on reset.	ne	aw	1
9	**Phantom Functions Enable**	1: enables a device to use unclaimed functions as Phantom Functions to extend the number of outstanding transaction identifiers. Fixed value 0, not supported.	ne	0	0
8	**Extended Tag Field Enable**	1: enables a device to use an 8-bit Tag field as a requester. Fixed value 0, not supported.	ne	0	0

Bit	Name	Description	Write	Read	Reset value
7:5	**Max_Payload_Size**	This field sets the maximum TLP payload size for the device. As a receiver, the device must handle TLPs as large as the set value; as transmitter, the device must not generate TLPs exceeding the set value. Permissible values for programming are indicated by the Max_Payload_Size supported in the Device Capabilities register. Defined encodings for this field are: 0b000: 128 B maximum payload size. Device supports only 128 Bytes.	yes	aw	0
4	**Enable Relaxed Ordering**	1: The device is permitted to set the Relaxed Ordering bit in the Attributes field of initiated transactions that do not require strong write ordering. Fixed value 0.	ne	0	0
3	**Unsupported Request Reporting Enable**	This bit controls reporting of Unsupported Requests when set.	yes	aw	0
2	**Fatal Error Reporting Enable**	This bit controls reporting of fatal errors.	yes	aw	0
1	**Non-Fatal Error Reporting Enable**	This bit controls reporting of non-fatal errors.	yes	aw	0
0	**Correctable Error Reporting Enable**	This bit controls reporting of correctable errors.	yes	aw	0

3.2.4.27 Device Status Register

Address: 0x0ea Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Device Status Register					
15:6	**Reserved**				
5	**Transactions Pending**	1: indicates that a device has issued Non-Posted Requests which have not been completed. A device reports this bit cleared only when all Completions for any outstanding Non-Posted Requests have been received.	ne	value	
4	**AUX Power Detected**	1: Devices that require AUX power report this bit as set if AUX power is detected by the device.	ne	value	
3	**Unsupported Request Detected**	1: indicates that the device received an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register. Default value is 0.	sh	value	0
2	**Fatal Error Detected**	1: indicates status of fatal errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control register. For devices supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the correctable error mask register. Default value is 0.	sh	value	0

Bit	Name	Description	Write	Read	Reset value
1	**Non-Fatal Error Detected**	1: indicates status of nonfatal errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control register. For devices supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the correctable error mask register. Default value is 0.	sh	value	0
0	**Correctable Error Detected**	1: indicates status of correctable errors detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control register. For devices supporting Advanced Error Handling, errors are logged in this register regardless of the settings of the correctable error mask register. Default value is 0.	sh	value	0

3.2.4.28 Link Capabilities Register

Address: 0x0ec Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Link Capabilities Register					
31:24	**Port Number**	This field indicates the PCI Express port number for the given PCI Express Link. Hardware Initialized.	ne	value	00
23:18	**Reserved**				

Bit	Name	Description	Write	Read	Reset value
17:15	**L1 Exit Latency**	This field indicates the L1 exit latency for the given PCI Express Link. The value reported indicates the length of time this Port requires to complete transition from L1 to L0. Defined encodings are: 0b000: Less than 1 µs 0b001: 1 µs to less than 2 µs 0b010: 2 µs to less than 4 µs 0b011: 4 µs to less than 8 µs 0b100: 8 µs to less than 16 µs 0b101: 16 µs to less than 32 µs 0b110: 32 µs up to 64 µs 0b111: More than 64 µs	ne	0b111	0b111
14:12	**L0s Exit Latency**	This field indicates the L0s exit latency for the given PCI Express Link. The value reported indicates the length of time this Port requires to complete transition from L0s to L0. Defined encodings are: 0b000: Less than 64 ns 0b001: 64 ns to less than 128 ns 0b010: 128 ns to less than 256 ns 0b011: 256 ns to less than 512 ns 0b100: 512 ns to less than 1 µs 0b101: 1 µs to less than 2 µs 0b110: 2 µs up to 4 µs 0b111: Reserved	ne	0b010	0b010
11:10	**Active State Link PM Support**	This field indicates the level of active state power management supported on the given PCI Express Link. Defined encodings are: 0b00: Reserved 0b01: L0s entry supported 0b10: Reserved 0b11: L0s and L1 supported	ne	0b01	0b01

Bit	Name	Description	Write	Read	Reset value
9:4	**Maximum Link Width**	This field indicates the maximum width of the given PCI Express Link. Defined encodings are: 0b000000: Reserved	ne	x01 x01	x01 x01
3:0	**Maximum Link Speed**	This field indicates the maximum link speed of the given PCI Express Link. Defined encodings are: 0b0001 2.5 Gb/s link. All other encodings are reserved.	ne	0x01	0x01

3.2.4.29 Link Control Register

Address: 0x0f0 Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Link Control Register					
15:8	**Reserved**				
7	**Extended Sync**	1: forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0. This mode provides external devices monitoring the link time to achieve bit and symbol lock before the link enters L0 state and resumes communication.	yes	aw	0
6	**Common Clock Configuration**	1: indicates that this component and the component at the opposite end of this link are operating with a distributed common reference clock. 0: indicates that this component and the component at the opposite end of this link are operating with asynchronous reference clock. Components utilize this common clock configuration information to report the correct L0s and L1 Exit Latencies.	yes	aw	0
5:4	**Reserved**				

Bit	Name	Description	Write	Read	Reset value
3	**Read Completion Boundary (RCB)**	It determines the naturally aligned address boundaries on which a Read Request may be serviced with multiple Completions. Encodings are: 0b0: 64 bytes 0b1: 128 bytes Device supports only 128 bytes	yes	aw	1
2	**Reserved**				
1:0	**Active State Link PM Control**	This field controls the level of active state PM supported on the given PCI Express Link. Defined encodings are: 0b00: Disabled 0b01: L0s Entry Supported 0b10: Reserved 0b11: L0s and L1 Entry Supported	yes	aw	0

3.2.4.30 Link Status Register

Address: 0x0f2 Width [bit]: 16

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Link Status Register					
15:13	**Reserved**				
12	**Slot Clock Configuration**	1: indicates that the component uses the same physical reference clock that the platform provides on the connector. 0: 25 MHz reference clock. If the device uses an independent clock irrespective of the presence of a reference on the connector. Hardware initialized	ne	1	1

Bit	Name	Description	Write	Read	Reset value
11	**Link Training**	This read-only bit indicates that Link training is in progress; hardware clears this bit once Link training is complete. This field is not applicable and reserved for endpoint devices and Upstream Ports of Switches. For debug purpose, not used for Yukon EC	ne	value	
10	**Training Error**	This read-only bit indicates that a Link training error occurred. This field is not applicable and reserved for endpoint devices and Upstream Ports of Switches. This bit is cleared by hardware upon successful training of the Link to the L0 Link state. For debug purpose, not used for Yukon EC	ne	value	
9:4	**Negotiated Link Width**	This field indicates the negotiated width of the given PCI Express Link. Defined encodings are: 0b000001: x1 All other encodings are reserved.	ne	value	
3:0	**Link Speed**	This field indicates the negotiated Link speed of the given PCI Express Link. Defined encodings are: 0b0001: 2.5 Gb/s PCI Express Link All other encodings are reserved.	ne	value	

3.2.4.31 Advanced Error Reporting Enhanced Capability Header

Address: 0x0100Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Advanced Error Reporting Enhanced Capability Header					
31:20	**Next Capability Offset**	This field is a PCI-SIG defined ID number that indicates the nature and format of the extended capability.	ne	0	0
19:16	**Capability Version**	This field is a PCI-SIG defined version number that indicates the version of the capability structure present.	ne	0x01	0x01
15:0	**PCI Express Extended Capability ID**	This field contains the offset to the next PCI Express capability structure.	ne	0x01	0x01

3.2.4.32 Uncorrectable Error Status/Mask/Severity Register

Address: 0x0104:0x010c Width [bit]:3 x 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Uncorrectable Error Status Register					
31:21	**Reserved**				
20	**Unsupported Request Error**	1: indicates - Received unsupported TLP type or - Received unsupported message codes or - Failed address decoding on received TLP. Reset by PEX Sticky Reset	sh	value	0
19	**ECRC Error**	Not supported, fixed value 0.	ne	0	0

Bit	Name	Description	Write	Read	Reset value
18	**Malformed TLP**	1: indicates - Received TLP with data payload size larger than 128 B (Max_Payload_Size) or - Received TLP with undefined Type field value or - Received TLP with illegal Format field value or - Received TLP with data payload size different from expected according to the length field or - Received TLP with uninitialized VC (different than VC-0) or - Received request with address/length combination crossing 4 KB boundary or - Received TLP with TD = 1 but without TLP digest. Reset by PEX Sticky Reset	sh	value	0
17	**Receiver Overflow**	Not supported, fixed value 0.	ne	0	0
16	**Unexpected Completion**	1: indicates received unexpected completion TLP. Completion does not correspond to one of the outstanding Non-Posted requests. Reset by PEX Sticky Reset	sh	value	0
15	**Completer Abort**	Not supported, fixed value 0.	sh	0	0
14	**Completion Timeout**	1: indicates outstanding Non-Posted request to PCI-Express has expired. Reset by PEX Sticky Reset	sh	value	0
13	**Flow Control Protocol Error**	1: indicates Flow Control Protocol Error Status. Reset by PEX Sticky Reset	sh	0	0
12	**Poisoned TLP**	1: indicates poisoned TLP received. Reset by PEX Sticky Reset	sh	value	0
11:5	**Reserved**				
4	**Data Link Protocol Error**	1: indicates reception of an Acknowledge with out of range AckNak_Seq_Num (Ack/Nak Sequence Number). Reset by PEX Sticky Reset	sh	value	0
3:1	**Reserved**				
0	**Training Error**	Not supported, fixed value 0.	sh	0	0

Bit	Name	Description	Write	Read	Reset value
Uncorrectable Error Mask Register					
31:21	**Reserved**				
20	**Unsupported Request Error**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
19	**ECRC Error**	Not supported, fixed value 0.	ne	0	0
18	**Malformed TLP**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
17	**Receiver Over-flow**	Not supported, fixed value 0.	ne	0	0
16	**Unexpected Completion**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
15	**Completer Abort**	Not supported, fixed value 0.	ne	0	0
14	**Completion Tim-eout**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
13	**Flow Control Protocol Error**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
12	**Poisoned TLP**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
11:5	**Reserved**				
4	**Data Link Proto-col Error**	1: not send error message to RC, not lock First Error Pointer, not logged in the Header Log register. Reset by PEX Sticky Reset	yes	aw	0
3:1	**Reserved**				
0	**Training Error**	Not supported, fixed value 0.	ne	0	0

Bit	Name	Description	Write	Read	Reset value
Uncorrectable Error Severity Register					
31:21	**Reserved**				
20	**Unsupported Request Error**	Non-Fatal Reset by PEX Sticky Reset	yes	aw	0
19	**ECRC Error Severity**	Not supported, fixed value 0.	ne	0	0
18	**Malformed TLP**	Fatal Reset by PEX Sticky Reset	yes	aw	1
17	**Receiver Over-flow**	Fatal Not supported, fixed value 1.	ne	1	1
16	**Unexpected Completion**	Non-Fatal Reset by PEX Sticky Reset	yes	aw	0
15	**Completer Abort**	Non-Fatal Not supported, fixed value 0.	yes	aw	0
14	**Completion Tim-eout**	Non-Fatal Reset by PEX Sticky Reset	yes	aw	0
13	**Flow Control Protocol Error**	Non-Fatal Reset by PEX Sticky Reset	yes	aw	1
12	**Poisoned TLP**	Non-Fatal Reset by PEX Sticky Reset	yes	aw	0
11:5	**Reserved**				
4	**Data Link Proto-col Error**	Fatal Reset by PEX Sticky Reset	yes	aw	1
3:1	**Reserved**				
0	**Training Error**	Not supported, fixed value 1.	ne	1	1

3.2.4.33 Correctable Error Status/Mask Register

Address: 0x0110:0x0114Width [bit]:2 x 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Correctable Error Status Register					
31:13	**Reserved**				
12	**Replay Timer Timeout**	1: indicates replay timer expired, REPLAY_NUM did not rollover. Reset by PEX Sticky Reset	sh	value	0
11:9	**Reserved**				
8	**REPLAY_NUM Rollover**	1: indicates 4 consecutive replays were transmitted. Reset by PEX Sticky Reset	sh	value	0
7	**Bad DLLP**	1: indicates - LCRC Error detected in received DLLP or - DLLP is larger than 6 B. Reset by PEX Sticky Reset	sh	value	0
6	**Bad TLP**	1: indicates - LCRC Error detected in received TLP or - Sequence number error detected in received TLP. Reset by PEX Sticky Reset	sh	value	0
5:1	**Reserved**				
0	**Receiver Error**	1: indicates - Overflow/Underrun or - 8 B/10 B decode error or - Disparity Error or - Framing Error - STP or SDP without END or EDB to previous frame or - Framing error - unexpected K code (special symbols for framing and link management, refer to PCI Express Base specification) in a middle of a frame (not END or EDB). Reset by PEX Sticky Reset	sh	value	0

Bit	Name	Description	Write	Read	Reset value
Correctable Error Mask Register					
31:13	**Reserved**				
12	**Replay Timer Timeout**	1: not send error message to RC Reset by PEX Sticky Reset	yes	aw	0
11:9	**Reserved**				
8	**REPLAY_NUM Rollover**	1: not send error message to RC Reset by PEX Sticky Reset	yes	aw	0
7	**Bad DLLP**	1: not send error message to RC Reset by PEX Sticky Reset	yes	aw	0
6	**Bad TLP**	1: not send error message to RC Reset by PEX Sticky Reset	yes	aw	0
5:1	**Reserved**				
0	**Receiver Error**	1: not send error message to RC	yes	aw	0

3.2.4.34 Advanced Error Capabilities and Control Register

Address: 0x0118Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Advanced Error Capabilities and Control Register					
31:9	**Reserved**				
8	**ECRC Check Enable**	1: enables ECRC checking. Not supported, fixed value 0.	ne	0	0
7	**ECRC Check Capable**	1: indicates that the device is capable of checking ECRC. Not supported, fixed value 0.	ne	0	0
6	**ECRC Generation Enable**	1: enables ECRC generation. Not supported, fixed value 0.	ne	0	0
5	**ECRC Generation Capable**	1: indicates that the device is capable of generating ECRC. Not supported, fixed value 0.	ne	0	0
4:0	**First Error Pointer**	The First Error Pointer is a read-only register that identifies the bit position of the first unmasked error reported in the Uncorrectable Error Status register. Note that the reset value 0x1f corresponds to the bit position of the reserved register bit 31 in Uncorrectable Error Status Register. Reset by PEX Sticky Reset	ne	value	0x1f

3.2.4.35 Header Log Register

Address: 0x011c:0x0128Width [bit]:4 x 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Header Log Register					
127:0	**Header Log**	Header of TLP associated with an error if this error is the first unmasked uncorrectable error detected. Reset by PEX Sticky Reset	ne	value	0

3.2.4.36 Transaction Layer Control Register

Address: 0x0200Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Transaction Layer Control Register					
31:5	**Reserved**				
1:0	**Max_outstand**	Maximum outstanding NP requests. 0x0: 1 request 0x1: 2 requests 0x2: 4 requests 0x3: 8 requests	yes	aw	0x03

3.2.4.37 Transaction Layer Status Register

Address: 0x0204Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Transaction Layer Status Register					
31:0	**Reserved**				

3.2.4.38 Data Link Layer Control Register

Address: 0x0208Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Data Link Layer Control Register					
31:0	**Reserved**				

3.2.4.39 Data Link Layer Status Register

Address: 0x020cWidth [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
Data Link Layer Status Register					
31:0	**Reserved**				

3.2.4.40 PE Physical Layer Control Register

Address: 0x0210Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Physical Layer Control Register					
31:16	**Reserved**				
15:8	**N_FTS**	Number of FTS Ordered-Sets needed by the device for L0s exit to L0. Advertised in transmitted TSs.	yes	aw	0x14
7:0	**Reserved**				

3.2.4.41 PE Physical Layer Status Register

Address: 0x0214Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Physical Layer Status Register					
31:0	**Reserved**				

3.2.4.42 PE Completion Timeout Register

Address: 0x0220Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Completion Timeout Register					
31:16	**Reserved**				
15:0	**CmpTOThrshld**	Completion Timeout Threshold. Controls the size of the completion timeout time interval. NOTE: Time scale: 256 * symbol_time = 1 µs 0x0: Disabled. No timeout mechanism on NP TLPs. Minimum Value: 40 (40 µs) Maximum Value: 25 K (25 ms)	yes	aw	0x2710 (10000 or 10 ms)

3.2.4.43 PE Flow Control Register

Address: 0x0224Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Flow Control Register					
31:24	**PHInitFC**	Posted Headers Flow Control Credit Initial Value.	yes	aw	0x02
23:16	**NPHInitFC**	Non-Posted Headers Flow Control Credit Initial Value.	yes	aw	0x02
15:8	**CHInitFC**	Completion Headers Flow Control Credit Initial Value.	yes	aw	0
7:0	**FCUpdateTO**	Flow Control Update Timeout. Controls the Flow Control update interval period. NOTE: Time scale: 64 * symbol_time = 250 ns. 0x0: Disabled. No timeout mechanism on update FC. Minimum Value: 120 (30 µs) Maximum Value: 180 (45 µs)	yes	aw	0x078 (120 or 30 µs)

3.2.4.44 PE Ack Timer for 1x Link

Address: 0x0228Width [bit]: 32

Reloadable out of the SPI Flash Memory or TWSI EEPROM.

Bit	Name	Description	Write	Read	Reset value
PE Ack Timer for 1x Link					
31:16	**AckRplyTOX1**	Ack Replay Timer Timeout Value for x1. NOTE: Timescale: symbol_time = 4 ns Minimum Value: 711 (2.84 µs) Maximum Value: 64K-1 (262 µs)	yes	aw	0x0320 (800)
15:0	**AckLatTOX1**	Ack Latency Timer Timeout Value for x1 link. Used when PHY link width auto-negotiation result is x1. NOTE: Time scale: symbol_time = 4 ns Minimum Value: 4 (16 ns) Maximum Value: 237 (948 ns)	yes	aw	0x04

3.3 Control Register File

The Control Register File demonstrates all registers accessible to the host, independent of their specific physical location.

It may be mapped in the 32-bit I/O space as well as in the 32-bit memory space depending on **Base Address Register (1st)** and **Base Address Register (2nd).**

Mapping to the I/O space may be disabled by **En IO Mapping**.

Write operations to reserved or not implemented registers are completed normally on the bus and the data is discarded.

Read operations to reserved or not implemented registers are completed normally on the bus, read data is undefined.

If **SW Reset** is set, it is not recommended to access the **Control Register File** except for accessing the **Control Register** (all internal or external devices are in reset state).
Write operations are completed normally on the bus and the data is discarded.
Read operations are completed normally on the bus and a data value of 0 is returned.

In order to prevent setup and hold time violations, all signals and/or events routed through read registers are synchronized to **CLK** (PCI).

All signals and/or events routed through **INTA#** are glitch free and, except for those directly related to **CLK** (PCI), independent of **CLK** (PCI).

3.3.1 Overview and Address Map

The table below depicts the layout of the Control Register File.

If the Control Register File is mapped into the memory space, it covers a memory range of 16 KB. All registers may be accessed directly.

If the Control Register File is mapped into the I/O space, it covers an I/O range of 256 byte. The 16 KB range of the Control Register File is segmented into 128 128-byte blocks.

Block 0 is mapped permanently into the lower half of the 256 byte I/O range.

Block 0 - 127 are mapped into the upper half of the 256 byte I/O range as defined by the Register Address Port (RAP). As a side effect, register addressing with the RAP may also be used, if the Control Register File is mapped into the memory space

Column 'STI' marks registers residing behind the same Target Interface (all registers within the same row separation line).

Byte<3>	**Byte<2>**	**Byte<1>**	**Byte<0>**	**Address**	**STI**	**Block**
Reserved	Reserved	Reserved	Register Address Port (RAP)	0x0000		0
Power Control	Control/Status Register			0x0004		
Interrupt Source Register				0x0008		
Interrupt Mask Register				0x000c		
Interrupt Hardware Error Source Register				0x0010		
Interrupt Hardware Error Mask Register				0x0014		
Special Interrupt Source Register 1				0x0018		
Special Interrupt Source Register 2				0x001c		
Special Interrupt Source Register 3				0x0020		
Enter Interrupt Service Routine Register				0x0024		
Leave Interrupt Service Routine Register				0x0028		
Interrupt Control Register				0x002c		
Reserved (legacy)				0x0030 : 0x005c		
SPI Flash Memory Control Register				0x0060		
SPI Flash Memory Address Register				0x0064		
SPI Flash Memory Data Register				0x0068		
SPI Flash Memory Vendor/Device ID Register				0x006c		
SPI Flash Memory Loader Configuration Register				0x0070		
SPI Flash Memory VPD Configuration Register				0x0074		
SPI Flash Memory Opcode 1 Register				0x0078		
SPI Flash Memory Opcode 2 Register				0x007c		
Block Window Register <31:0> Note: If RAP = 1, read values are ZERO, write cycles have no effect				0x0080 : 0x00fc		1

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
MAC Addr_1<3>	MAC Addr_1<2>	MAC Addr_1<1>	MAC Addr_1<0>	0x0100		2
Reserved	Reserved	MAC Addr_1<5>	MAC Addr_1<4>	0x0104		
Reserved	Reserved	MAC Addr_2<5>	MAC Addr_2<4>	0x010c		
MAC Address for maintenance						
MAC Addr_3<3>	MAC Addr_3<2>	MAC Addr_3<1>	MAC Addr_3<0>	0x0110		
Reserved	Reserved	MAC Addr_3<5>	MAC Addr_3<4>	0x0114		
Chip ID	Chip Revision	PMD Type	Connector Type	0x0118		
Eprom<3>	Eprom<2> HW Resources	Eprom<1> Clock Gating Register	Eprom<0> RAM Size	0x011c		
Reserved	Clock Divider value	Clock Divider Control		0x120		
Reserved (legacy)				0x0124 : 0x012c		
IRQ Timer						
IRQ Timer Init Value				0x0130	STI	
IRQ Timer				0x0134		
Reserved	Reserved	IRQ Timer Test	IRQ Timer Control	0x0138		
Reserved				0x013c		
IRQ Moderation Timer						
IRQ Moderation Timer Init Value				0x0140	STI	
IRQ Moderation Timer				0x0144		
Reserved	Reserved	IRQ Moderation Timer Test	IRQ Moderation Timer Control	0x0148	STI	
Interrupt Moderation Mask Register				0x014c		
Interrupt Hardware Error Moderation Mask Register				0x0150		
Reserved				0x0154		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
						2
Reserved		Test Control Reg 2	Test Control Reg 1	0x0158		
General Purpose IO Register				0x015c		
TWSI (HW) Control Register				0x0160		
TWSI (HW) Data Register				0x0164		
TWSI (HW) IRQ Register				0x0168		
TWSI (SW) Register				0x016c		
PEX PHY Address Register		PEX PHY Data Register		0x0170		
Reserved				0x0174		
Reserved (legacy)				0x0178		
Reserved				0x017c		
RAM Random Registers						3
RAM Address				0x0180	STI	
RAM Data Port/lower dword				0x0184		
RAM Data Port/upper dword				0x0188		
Reserved				0x018c		
RAM Interface Registers						
Timeout Init Value 3 (Read SM Rx1)	Timeout Init Value 2 (Write SM Tx/s1)	Timeout Init Value 1 (Write SM Tx/a1)	Timeout Init Value 0 (Write SM Rx1)	0x190	STI	
Reserved	Reserved)	Timeout Init Value 5 (Read SM Tx/s1)	Timeout Init Value 4 (Read SM Tx/a1)	0x0194		
Reserved				0x0198		
Reserved			Timeout Timer	0x019c		
Reserved	Timer Test	RAM Interface Control		0x01a0		
Reserved				0x01a4 : 0x01fc		3

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Transmit Arbiter						4
Interval Timer Init Value				0x0200	STI	
Interval Timer				0x0204		
Limit Counter Init Value				0x0208		
Limit Counter				0x020c		
Reserved	Timer/ Counter Status	Timer/ Counter Test	Timer/ Counter Control	0x0210		
Reserved				0x0214 : 0x021c		
RSS Key						
RSS Key 0				0x0220		
RSS Key 1				0x0224		
RSS Key 2				0x0228		
RSS Key 3				0x022c		
Reserved				0x0230 : 0x027c		
Reserved				0x0294 : 0x029c		5
Reserved				0x02b0 : 0x02fc		
Reserved	Reserved	Reserved (legacy)		0x0300 : 0x037c		6
PCI Configuration Register File (lower half)				0x0380 : 0x03fc		7

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Receive Queue						8
Current Receive Descriptor						
Receive Buffer Control		Receive Buffer Byte Count		0x0400		
RSS Hash Checksum				0x0404		
Receive Buffer Address, lower dword				0x0408		
Receive Buffer Address, upper dword				0x040c		
Receive Status Word				0x0410		
Receive Timestamp				0x0414		
TCP/IP Checksum Extension						
TCP Checksum 2		TCP Checksum		0x0418		
Start position, TCP Checksum 2		Start position, TCP Checksum		0x041c		
Reserved		VLAN Tag		0x0420		
Reserved		Done Index		0x0424		
Request Address, lower dword				0x0428		
Request Address, upper dword				0x042c		
Reserved		Request Byte Count		0x0430		
BMU Control/Status Register				0x0434		
BMU Test Register				0x0438		
BMU State Machine Register				0x043c		
Reserved	FIFO Alignment	FIFO watermark		0x0440		
Reserved	FIFO read shadow level	FIFO read shadow pointer		0x0444		
Reserved	FIFO read level	Reserved	FIFO read pointer	0x0448		
FIFO write shadow level	FIFO write level	FIFO write shadow pointer	FIFO write pointer	0x044c		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
RX Prefetch Unit						8
Prefetch Control Register				0x0450		
Reserved		Last Index		0x0454		
List Start Address, low				0x0458		
List Start Address, high				0x045c		
Reserved		Get Index		0x0460		
Reserved		Put Index		0x0464		
Reserved				0x0468 : 0x046c		
FIFO write shadow pointer		Reserved	FIFO write pointer	0x0470		
Reserved			FIFO read pointer	0x0474		
Master Request nbytes		Reserved	FIFO watermark	0x0478		
FIFO shadow level		Reserved	FIFO level	0x047c		
Reserved						9
Reserved	Reserved	Reserved		0x0500 : 0x05fc		10 : 11

Byte<3>	**Byte<2>**	**Byte<1>**	**Byte<0>**	**Address**	**STI**	**Block**
Asynchronous Transmit Queue						13
Current Transmit Descriptor						
Transmit Buffer Control		Transmit Buffer Byte Count		0x0680		
Reserved				0x0684		
Transmit Buffer Address, lower dword				0x0688		
Transmit Buffer Address, upper dword				0x068c		
Transmit Status Word				0x0690		
TCP/IP Checksum Extension						
Reserved		TCP Sum Init Value		0x0694		
TCP Sum Start		TCP Sum Write		0x0698		
Reserved				0x069c		
Reserved		VLAN Tag		0x06a0		
Reserved		Done Index		0x06a4		
Request Address, lower dword				0x06a8		
Request Address, upper dword				0x06ac		
Reserved		Request Byte Count		0x06b0		
BMU Control/Status Register				0x06b4		
BMU Test Register				0x06b8		
BMU State Machine Register				0x06bc		
Reserved	FIFO Alignment	FIFO watermark		0x06c0		
Reserved	FIFO write shadow level	FIFO write shadow pointer		0x06c4		
Reserved	FIFO write level	Reserved	FIFO write pointer	0x06c8		
Reserved	FIFO read level	Reserved	FIFO read pointer	0x06cc		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
TXA Prefetch Unit						13
Prefetch Control Register				0x06d0		
Reserved		Last Index		0x06d4		
List Start Address, low				0x06d8		
List Start Address, high				0x06dc		
Reserved		Get Index		0x06e0		
Reserved		Put Index		0x06e4		
Reserved				0x06e8 : 0x06ec		
FIFO write shadow pointer		Reserved	FIFO write pointer	0x06f0		
Reserved			FIFO read pointer	0x06f4		
Master Request nbytes		Reserved	FIFO watermark	0x06f8		
FIFO shadow level		Reserved	FIFO level	0x06fc		

Byte<3>	**Byte<2>**	**Byte<1>**	**Byte<0>**	**Address**	**STI**	**Block**
Receive Rambuffer						16
Receive Rambuffer Start Address				0x0800	STI	
Receive Rambuffer End Address				0x0804		
Receive Buffer Write Pointer				0x0808		
Receive Buffer Read Pointer				0x080c		
Receive Rambuffer Upper Threshold/Pause Packets				0x0810		
Receive Rambuffer Lower Threshold/Pause Packets				0x0814		
Receive Rambuffer Upper Threshold/High Priority				0x0818		
Receive Rambuffer Lower Threshold/High Priority				0x081c		
Receive Rambuffer Packet Counter				0x0820		
Receive Rambuffer Level				0x0824		
Receive Rambuffer Control/Test				0x0828		
Reserved				0x082c : 0x087c		
Reserved						17

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Reserved						18
Reserved				0x0900	STI	
Reserved				0x0904		
Reserved				0x0908		
Reserved				0x090c		
Reserved				0x0910		
Reserved				0x0914		
Reserved				0x0918		
Reserved				0x091c		
Reserved				0x0920		
Reserved				0x0924		
Reserved				0x0928		
Reserved				0x092c : 0x093c		
Reserved						
Reserved				0x0940	STI	
Reserved				0x0944		
Reserved				0x0948		
Reserved				0x094c		
Reserved				0x0950 : 0x095c		
Reserved				0x0960		
Reserved				0x0964		
Reserved				0x0968		
Reserved 0x096c : 0x097c						

Byte<3>	**Byte<2>**	**Byte<1>**	**Byte<0>**	**Address**	**STI**	**Block**
Asynchronous Transmit Rambuffer						21
Transmit Rambuffer Start Address				0x0a80	STI	
Transmit Rambuffer End Address				0x0a84		
Transmit Buffer Write Pointer				0x0a88		
Transmit Buffer Read Pointer				0x0a8c		
Reserved				0x0a90 : 0x0a9c		
Transmit Rambuffer Packet Counter				0x0aa0		
Transmit Rambuffer Level				0x0aa4		
Transmit Rambuffer Control/Test				0x0aa8		
Reserved				0x0aac : 0x0afc		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Receive MAC FIFO						24
Reserved (legacy)				0x0c00 : 0x0c3c		
Receive MAC FIFO End Address				0x0c40	STI	
Receive MAC FIFO Almost Full Threshold				0x0c44		
Receive MAC FIFO Control/Test				0x0c48		
Receive MAC FIFO Flush Mask				0x0c4c		
Receive MAC FIFO Flush Threshold				0x0c50		
Receive Truncation Threshold				0x0c54		
Reserved				0x0c58		
Receive VLAN Type Register				0x0c5c		
Receive MAC FIFO Write Pointer				0x0c60		
Reserved				0x0c64		
Receive MAC FIFO Write Level				0x0c68		
Reserved				0x0c6c		
Receive MAC FIFO Read Pointer				0x0c70		
Reserved				0x0c74		
Receive MAC FIFO Read Level				0x0c78		
Reserved				0x0c7c		
Reserved						25
Reserved (legacy)				0x0c80 : 0x0cbc		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Transmit MAC FIFO						26
Reserved (legacy)				0x0d00 : 0x0d28		
Reserved				0x0d2c : 0x0d3c		
Transmit MAC FIFO End Address				0x0d40	STI	
Transmit MAC FIFO Almost Empty Threshold				0x0d44		
Transmit MAC FIFO Control/Test				0x0d48		
Reserved				0x0d4c : 0x0d58		
Transmit VLAN Type Register				0x0d5c		
Transmit MAC FIFO Write Pointer				0x0d60		
Transmit MAC FIFO Write Shadow Pointer				0x0d64		
Transmit MAC FIFO Write Level				0x0d68		
Reserved				0x0d6c		
Transmit MAC FIFO Read Pointer				0x0d70		
Transmit MAC FIFO Restart Pointer				0x0d74		
Transmit MAC FIFO Read Level				0x0d78		
Reserved				0x0d7c		
Reserved				0x0d80 | 0x0dfc		27

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Descriptor Poll Timer						28
Timer Init Value				0x0e00	STI	
Timer				0x0e04		
Reserved	Timer Test	Reserved	Timer Control	0x0e08		
Reserved				0x0e0c		
Timestamp Timer						
Reserved				0x0e10	STI	
Timer				0x0e14		
Reserved	Timer Test	Reserved	Timer Control	0x0e18		
Reserved				0x0e1c		
Polling Unit						
Poll Control				0x0e20		
Reserved		List Last Index		0x0e24		
List Start Address, low				0x0e28		
List Start Address, high				0x0e2c		
Reserved				0x0e30 : 0x0e3c		
Reserved						
Reserved						
Reserved				0xe40		
Reserved				0xe44		
Reserved				0x0e48 : 0x0e5c		
Reserved						
Reserved				0xe60		
Reserved				0xe64		
Reserved				0xe68		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Host Status and Command Register				0xe6c		28
Data Register 1				0xe70		
Data Register 2				0xe74		
Data Register 3				0xe78		
Data Register 4				0xe7c		

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Status BMU						29
Status BMU Control				0x0e80		
Reserved		Last Index		0x0e84		
List Start Address, low				0x0e88		
List Start Address, high				0x0e8c		
Reserved		TXA1 Report Index		0x0e90		
Reserved		Reserved		0x0e94		
Reserved		TX Index Threshold		0x0e98		
Reserved		Put Index		0x0e9c		
FIFO control/status						
Reserved			FIFO write pointer	0x0ea0		
Reserved			FIFO read pointer	0x0ea4		
Reserved			FIFO level	0x0ea8		
Reserved		FIFO ISR watermark	FIFO watermark	0x0eac		
Level Timer						
Level Timer Init Value				0x0eb0	STI	
Level Timer Counter				0x0eb4		
Reserved	Level Timer Test	Reserved	Level Timer Ctrl	0x0eb8		
Reserved				0x0ebc		
TX Timer						
TX Timer Init Value				0x0ec0	STI	
TX Timer Counter				0x0ec4		
Reserved	TX Timer Test	Reserved	TX Timer Control	0x0ec8		
Reserved				0x0ecc		
ISR Timer						
ISR Timer Init Value				0x0ed0	STI	
ISR Timer Counter				0x0ed4		
Reserved	ISR Timer Test	Reserved	ISR Timer Ctrl	0x0ed8		
Reserved				0x0edc		
Reserved				0x0ee0 0x0efc		

Byte<3>	**Byte<2>**	**Byte<1>**	**Byte<0>**	**Address**	**STI**	**Block**
MAC and PHY Control Registers						30
Reserved			MAC Control	0x0f00	STI	
PHY MUX Register			PHY Control	0x0f04		
Reserved			MAC Interrupt Source	0x0f08		
Reserved			MAC Interrupt Mask	0x0f0c		
Reserved			Link Control	0x0f10		
Reserved				0x0f14 : 0xf1c		
Wake On LAN Control Registers						
Match Result	Match Control	WOL Control/Status		0x0f20	STI	
MAC Address Low				0x0f24		
Reserved	PME Match Enable	MAC Address High		0x0f28		
Reserved			Pattern Read Pointer	0x0f2c		
Pattern 3 Length	Pattern 2 Length	Pattern 1 Length	Pattern 0 Length	0x0f30		
Reserved	Pattern 6 Length	Pattern 5 Length	Pattern 4 Length	0x0f34		
Pattern 3 Counter	Pattern 2 Counter	Pattern 1 Counter	Pattern 0 Counter	0x0f38		
Reserved	Pattern 6 Counter	Pattern 5 Counter	Pattern 4 Counter	0x0f3c		
Reserved				0x0f40 : 0xf7c		
Reserved				0x0f80 : 0x0ffc		31
Pattern RAM 256 words x 32 bit				0x1000 : 0x13fc		32 : 39
Pattern RAM 256 words x 32 bit				0x1400 : 0x18fc		40 : 49

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
				0x1900 : 0x1984		50 : 51
Reserved				0x1988 : 0x19fc		
Reserved				0x1a00 : 0x1bfc		52 : 55
Configuration Register File				0x1c00 : 0x1e7c		56 : 60
Reserved				0x1e80 : 0x1ffc		61 : 63
Reserved	Reserved	Reserved (legacy)		0x2000 : 0x27fc		64 : 79
Reserved	Reserved	GMAC Register <0x00> (registers according to GMAC datasheet)		0x2800		80 : 85
Reserved	Reserved	GMAC Register <0x01>		0x2804		
Reserved	Reserved	GMAC Register <0x02> : GMAC Register <0x96>		0x2808 : 0x2a58		
Reserved	Reserved	GMAC Register <0x97>		0x2a5c		
Reserved				0x2a60 : 0x2afc		
Reserved				0x2b00 : 0x2ffc		86 : 95
Reserved	Reserved	Reserved (legacy)		0x3000 : 0x37fc		96 : 111

Byte<3>	Byte<2>	Byte<1>	Byte<0>	Address	STI	Block
Reserved				0x3800 : 0x3ffc		112 : 127

3.3.2 Registers

3.3.2.1 Register Address Port (RAP)

Address: 0x0000 Width [bit]: 8

Bit	Name	Description	Write	Read	Reset (SW)
Register Address Port (RAP)					
31:7	**Reserved**				
6:0	**RAP**	Specifies one out of blocks 0 to 127, which is mapped to the upper half of the 256 byte I/O range. 0: block 0 .. 0x7f: block 127	yes	aw	0

3.3.2.2 Control/Status Register

Address: 0x0004 Width [bit]: 24

Bit	Name	Description	Write	Read	Reset
Control/Status Register					
	Status				
23:18	**Reserved**				
17	**Vmain Available**	**Vmain Available** is the **Vmain** Pin of the PCI Bus used as input with a weak Pull-down. 0: no Vmain available 1: Vmain available	ne	value	value

Bit	Name	Description	Write	Read	Reset
16	**Vaux Available**	**Vaux Available** is the **Vaux** Pin of the PCI Bus used as input with a weak Pull-down. 0: no Vaux available 1: Vaux available	ne	value	value
	Commands				
15:14	**Reserved**				
13	**Reserved**	Reserved	exec	0b10	0
12	**Reserved**			0b01	1
11:10	**Reserved**		exec	0b10	0
				0b01	1
9	**LED<0>On**	LED On/Off Commonly used as indication for “Driver loaded”	exec	0b10	0
8	**LED<0> Off**			0b01	1
7	**Set IRQ SW**	Sets and clears Interrupt Request from SW	exec	0b10	0
6	**Clear IRQ SW**			0b01	1
5	**Stop Master Done**	As soon as the Master Statemachine is in the idle state after **Stop Master** is set, **Stop Master Done** is asserted. **Stop Master Done** is reset to 0 by resetting **Stop Master**.	ne	value	0
4	**Stop Master**	If **Stop Master** is set, all requests from the BMUs except for the one being serviced at the moment, are masked. The Master Statemachine reaches the idle state after the current request is serviced. **Stop Master** has to be reset by the SW after the BMUs are reset. If the BMUs are not reset, the PCI device resumes master action at the point, when it was interrupted by **Stop Master**.	yes	aw	0 (HW)
3	**Master Reset Clear**	Set/Clear **Master Reset.** If **Master Reset** is set, all devices related to the master interface (BMUs, FIFOs, State machines) are in their reset state. Executed, if appropriate bit is set to 1.	exec	0b10	0
2	**Master Reset Set**			0b01	1 (SW)

Bit	Name	Description	Write	Read	Reset
1	**SW Reset Clear**	Set/Clear **SW Reset.** Executed, if appropriate bit is set to 1. If **SW Reset** is set, all internal and external devices are in their reset state.	exec	0b10	0
0	**SW Reset Set**			0b01	1 (HW)

3.3.2.3 Power Control Register

Address: 0x0007 Width [bit]: 8

Bit	Name	Description	Write	Read	Reset (Power On)
Power Control Register					
7	**Switch Vaux Enable**	Switch Vaux Enable 1 = Output Switch Vaux pin enabled 0 = Output Switch Vaux pin disabled	exec	0b10	0
6	**Switch Vaux Disable**			0b01	1
5	**Switch VCC Enable**	Switch VCC Enable 1 = Output Switch VCC pin enabled 0 = Output Switch VCC pin disabled	exec	0b10	0
4	**Switch VCC Disable**			0b01	1
3	**Switch Vaux On**	Switch Vaux On/Off 1 = Power supply from Vaux pin on 0 = Power supply from Vaux pin off	exec	0b10	0
2	**Switch Vaux Off**			0b01	1
1	**Switch VCC On**	Switch VCC On/Off 1 = Power supply from VCC pins on 0 = Power supply from VCC pins off	exec	0b10	0
0	**Switch VCC Off**			0b01	1

3.3.2.4 Interrupt Source Register

Address: 0x0008 Width [bit]: 32

If set to ONE, interrupt is pending

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Source Register				
	General Interrupts				
31	**HW Interrupt**	1: At least one of the HW Interrupts occurred (**Interrupt HW Error Source Register**), which is not masked by the according Mask register. 0: No HW Interrupt active	ne	value	0
30	**Status BMU**	Interrupt on status burst	ne	value	0
29	**Reserved**	Reserved	ne	value	0
28	**Reserved**				
27	**IRQ Polling CHK**	Check Interrupt by the Polling Unit	ne	value	0
26	**IRQ TWSI Ready**	Interrupt on completion of TWSI transfer	ne	value	0
25	**IRQ SW**	Interrupt set by SW in Control Register	ne	value	0
24	**IRQ Timer**	Interrupt Timer	ne	value	0
23:16	**Reserved**				
	Interrupts				
7:5	**Reserved**				
4	**IRQ PHY**	Interrupt from PHY	ne	value	0
3	**IRQ MAC**	Interrupt from MAC	ne	value	0
2	**IRQ CHCK Rx**	Interrupt coding error of descriptor (Rx Queue)	ne	value	0
1	**Reserved**		ne	value	0
0	**IRQ CHCK TxA**	Interrupt Coding Error of descriptor (asynchronous Tx Queue)	ne	value	0

3.3.2.5 Interrupt Mask Register

Address: 0x000c Width [bit]: 32

Each bit position defines, if the dedicated interrupt is propagated to the internal interrupt line **irq**.
The enable bits have the same bit positions as in the **Interrupt Source Register.** Unused bit positions are treated like reserved.

If set to ONE, interrupt is enabled.

See also “**Special Interrupt Source Register “.**

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Mask Register				
31:0	**En IRQ xxx**	Enable Interrupt xxx	yes	aw	0

3.3.2.6 Interrupt HW Error Source Register

Address: 0x0010 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt HW Error Source Register				
	General Interrupts				
31:30	**Reserved**				
29	**IRQ Timestamp Timer Overflow**	Interrupt Timestamp Timer Overflow	ne	value	0
28	**IRQ Sensor**	Interrupt from sensor This external interrupt line is connected to the interrupt output of the Voltage/Temperature Sensor	ne	value	0
27	**IRQ Master Error**	Interrupt Master Error detected on master accesses Set, if **DATAPERR**, **RTABORT** or **RMABORT** are set.	ne	value	0
26	**IRQ Status**	Interrupt Status Exception Set, if **PERR, RMABORT, RTABORT** or **DATAPERR** are set.	ne	value	0

Bit	Name	Description	Write	Read	Reset (SW)
25	**IRQ PE**	PCI Express Interrupt Indication of occurrence of uncorrectable error(s) in PCI-Express mode	ne	value	0
24	**IRQ NO PE**	An error occurs in PCI-Express mode. Note: This is not a PCI-Express error!	ne	value	0
23:16	**Reserved**				
	HW Interrupts				
7:6	**Reserved**				
3	**IRQ Par MAC**	Interrupt Parity Error/MAC This interrupt is intended to indicate a panic event (hardware fault)	ne	value	0
2	**IRQ PAR Rx**	Interrupt Parity Error (RxQueue) This interrupt is intended to indicate a panic event (hardware fault)	ne	value	0
1			ne	value	0
0	**IRQ TCP Length TxA1**	Interrupt Length Mismatch (asynchronous Tx queue 1) with TCP segmentation	ne	value	0

3.3.2.7 Interrupt HW Error Mask Register

Address: 0x0014 Width [bit]: 32

Each bit position defines, if the dedicated interrupt is propagated to the Interrupt Line **INTA#**.
The enable bits have the same bit positions as in the **Interrupt HW Error Source Register.** Unused bit positions are treated like reserved.

Bits, which are dedicated to a potential second MAC are reserved, but their setting have no effect.

If set to ONE, interrupt is enabled.

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Hardware Error Mask Register				
31:0	**En IRQ xxx**	Enable Hardware Interrupt xxx	yes	aw	0

3.3.2.8 Special Interrupt Source Register 1

Address: 0x0018 Width [bit]: 32

This register mirrors the **Interrupt Source Register** with special functionality adapted to typical SW handling.
If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.

If the internal interrupt line **irq** is NOT asserted, the read value is 0.

If the internal interrupt line **irq** is asserted, reading the **Special Interrupt Source Register 1** masks all Interrupts. As a result the internal interrupt line **irq** is deasserted.

Bit positions are the same as in the **Interrupt Source Register.**

Bit	Name	Description	Write	Read	Reset (SW)
	Special Interrupt Source Register 1				
31:0	**IRQ xxx**	Interrupt xxx When the internal interrupt line **irq** is asserted, read masks all interrupts. As a result the internal interrupt line **irq** is deasserted.	ne	value or 0	0

3.3.2.9 Special Interrupt Source Register 2

Address: 0x001c Width [bit]: 32

This register mirrors the **Interrupt Source Register** with special functionality adapted to typical SW handling.
If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.

If the internal interrupt line **irq** is NOT asserted, the read value is 0.

If the internal interrupt line **irq** is asserted, reading the **Special Interrupt Source Register 2** masks all Interrupts and sets **isr status** flag **of Interrupt Control Register** to “ISR mode”. As a result the internal interrupt line **irq** is deasserted.

Bit positions are the same as in the **Interrupt Source Register.**

Bit	Name	Description	Write	Read	Reset (SW)
	Special Interrupt Source Register 2				
31:0	**IRQ xxx**	Interrupt xxx When the internal interrupt line **irq** is asserted, read masks all interrupts. As a result the internal interrupt line **irq** is deasserted. When **irq** is active, the **isr status** flag **of Interrupt Control Register** is set to “ISR mode”.	ne	value or 0	0

3.3.2.10 Special Interrupt Source Register 3

Address: 0x0020 Width [bit]: 32

This register mirrors the **Interrupt Source Register** with special functionality adapted to typical SW handling.

If the internal interrupt line **irq** is asserted, the read value is the same as in the **Interrupt Source Register**.

If the internal interrupt line **irq** is NOT asserted, the read value is 0.

Reading the **Special Interrupt Source Register 3** always masks all interrupts. As a result the internal interrupt line **irq** is deasserted.

If the internal interrupt line **irq** is asserted, **isr_status** flag is set to “ISR mode”.

Bit positions are the same as in the **Interrupt Source Register.**

Bit	Name	Description	Write	Read	Reset (SW)
	Special Interrupt Source Register 3				
31:0	**IRQ xxx**	Interrupt xxx Read masks all interrupts. As a result the internal interrupt line **irq** is deasserted. When **irq** is active, the **irq status** flag is set to “ISR mode”.	ne	value or 0	0

3.3.2.11 Enter Interrupt Service Routine Register

Address: 0x0024 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
	Enter Interrupt Service Routine Register				
31:0	**Enter ISR Reg**	Read: **isr_status** is set and all interrupts are masked. Value of Interrupt Source Register is returned Write: no effect	ne	value	0

3.3.2.12 Leave Interrupt Service Routine Register

Address: 0x0028 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
	Leave Interrupt Service Routine Register				
31:0	**Leave ISR Reg**	Read: **isr_status** is reset and the overall masking of interrupts is released. Value of Interrupt Source Register is returned Write: no effect	ne	value	0

3.3.2.13 Interrupt Control Register

Address: 0x002c Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Control Register				
31:4	**reserved**		ne	0	0
3	**isr_mask**	**isr_mask** flag value	ne	value	0
2	**isr_status**	**isr_status** flag value 1: “ISR mode” 0: “normal mode”	ne	value	0
1	**Leave ISR**	written to with 1: **isr_status** is reset and the overall masking of interrupts is released.	exec	0	0
0	**Enter ISR**	written to with 1: **isr_status** is set and all interrupts are masked.	exec	0	0

3.3.2.14 SPI Flash Memory Control Register

Address: 0x0060 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Control Register					
31	**Loader Start**	Start to load SPI configuration from loader start address, software can reload again using this bit. Start to load SPI configuration from the current address stored in the SPI Flash Memory Address Register. During the loading time software accesses to registers are hold until loading has been finished.	yes	0	0
30	**SPI Busy**	Status bit indicating that a command is in progress on the SPI	no	yes	0

Bit	Name	Description	Write	Read	Reset (HW)
29	**SPI VPD enable**	0: (default) VPD provided in TWSI EE-PROM 1: VPD is mapped to SPI Flash Memory. Enables VPD accesses to the SPI Flash Memory.	yes	yes	0
28	**RD ID Protocol**	SPI Flash Memories of different vendors need different protocols for reading the ID: 0: The ID is transferred immediately by the SPI Flash Memory after Read ID instruction is initiated (ATMEL Flash Memory). 1: Dummy Address write (3 zero bytes) should precede the initial Read ID instruction. (SST and ST SPI Flash Memories)	yes	aw	0
27:20	**Reserved**		yes	0	0
19	**Start SPI**	An SPI Instruction is initiated by writing a ‘1’ to this register bit.	exec	0	0
18:16	**Instruction**	Selection of one of the opcodes of the SPI Flash Memory opcode registers: 0: Opcode No Operation 1: Opcode Read 2: Opcode Read ID 3: Opcode Read Status Register 4: Opcode Write Enable 5: Opcode Write 6: Opcode Sector Erase 7: Opcode Chip Erase	yes	yes	0
15:8	**Reserved**		yes	0	0
7:0	**SPI device Status**	Contents of the status register from the SPI device. Register is updated only after executing the Read Status Register command by the SPI Flash Memory. During write access or erasing of SPI Flash Memory, software should continue initiate Read Status Register instruction and afterwards reading this register until the last bit- bit[0] is 0 before start of next write or read access.	no	yes	0x00

3.3.2.15 SPI Flash Memory Address Register

Address: 0x0064 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Address Register					
31:20	**Reserved**		yes	0	0
19:0	**SPI Flash Memory address**	SPI Flash Memory Address for read or write accesses to the SPI Flash Memory	yes	yes	0x00000

3.3.2.16 SPI Flash Memory Data Register

Address: 0x0068 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Data Register					
31:0	**SPI Flash Memory data**	SPI Flash Memory data for write accesses or result of read access to the SPI Flash Memory	yes	yes	0x00000

3.3.2.17 SPI Flash Memory Vendor/Device ID Register

Address: 0x006c Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Vendor/Device ID Register					
31:16	**Reserved**		yes	0	0

Bit	Name	Description	Write	Read	Reset (HW)
15:8	**Vendor-ID**	To identify the Vendor of the Flash each device has a special code which can be read out with the RDID instruction	no	yes	0x00
7:0	**Device-ID**	To identify the Flash Device each device has a special code which can be read out with the RDID instruction	no	yes	0x00

3.3.2.18 SPI Flash Memory Loader Configuration Register

Address: 0x0070 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Loader Configuration Register					
31:28	**Reserved**		yes	0	0
27:16	**Normal Loader start address**	Start Address of Configuration data within SPI Flash Memory After reset SPI Flash Memory loader starts to load configuration data from this address. Address is aligned to 256 Byte boundaries. These bits are the higher bits of the address and bits 7:0 are set to ‘0’ implicitly.	yes	yes	0x1f8
15:12	**Reserved**		yes	0	0
11:0	**PIG Loader start address**	Start Address of Configuration data within SPI Flash Memory when in Plug-in-Go mode During power save mode SPI Flash Memory loader will start to load configuration data from this address Address is aligned to 256 Byte boundaries. These bits are the higher bits of the address and bits 7:0 are set to ‘0’ implicitly.	yes	yes	0x1f0

3.3.2.19 SPI Flash Memory VPD Configuration Register

Address: 0x0074 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory VPD Configuration Register					
31:28	**Reserved**		yes	0	0
27:16	**VPD end address**	When VPD is mapped into SPI Flash Memory, VPD area ends at this address. Accesses beyond this end address are invalid accesses. Address is aligned to 256 Byte boundaries. These bits are the higher bits of the address and bits 7:0 are set to ‘0’ implicitely.	yes	yes	0x1c1
15:12	**Reserved**		yes	0	0
11:0	**VPD start address**	When VPD is mapped into SPI Flash Memory, VPD area starts at this address. Accesses below this start address are invalid accesses. Address is aligned to 256 Byte boundaries. These bits are the higher bits of the address and bits 7:0 are set to ‘0’ implicitely.	yes	yes	0x1c0

3.3.2.20 SPI Flash Memory Opcode 1 and 2 Register

Address: 0x0078, 0x007c Width [bit]: 2 x 32

The listed default values for the opcodes are valid for SST’s serial flash memories.

The values must be adapted by software when using SPI flash memories of other providers. Especially the opcodes of Sector Erase, Chip Erase and Read ID must be changed. All other opcodes are equal for the SPI Flash Memories supported by Yukon™ EC.

Bit	Name	Description	Write	Read	Reset (HW)
SPI Flash Memory Opcode 1 Register					

Bit	Name	Description	Write	Read	Reset (HW)
31:24	**Opcode Read Status**	Opcode for reading the status register of SPI Flash Memory. Result is stored within lower bits of SPI Flash Memory Control Register.	yes	aw	0
23:16	**Opcode Read ID**	Opcode for reading the Vendor/ Device ID out of SPI Flash Memory. Result is stored within SPI Flash Memory Vendor/ Device ID Register.	yes	aw	0
15:8	**Opcode Read**	Opcode for Read data out of SPI Flash Memory at address of SPI Flash Memory Address Register. Result is stored within SPI Flash Memory data register.	yes	aw	0
7:0	**Opcode No Operation**	Opcode for No Operation	yes	aw	0
SPI Flash Memory Opcode 2 Register					
31:24	**Opcode Chip Erase**	Opcode for SPI Flash Memory chip-erase command. During execution of this operation software should check SPI device status bit[0], SPI has not finished erasing until status bit[0] is 0.	yes	aw	0
23:16	**Opcode Sector Erase**	Opcode for a SPI Flash Memory sector-erase command. During execution of this operation software should check SPI device status bit[0], SPI has not finished erasing until status bit[0] is 0.	yes	aw	0
15:8	**Opcode Write**	Opcode for Writing data of SPI Flash Memory data register into SPI Flash Memory at address of SPI Flash Memory Address Register.	yes	aw	0
7:0	**Opcode Write Enable**	Opcode for the WREN command which sets the write enable latch. After execution of this command a Sector Erase, Chip Erase or Program command should follow.	yes	0	0

3.3.2.21 Block Window

Address: 0x0080.. 0x00fc Width [bit]: 128 x 32

One block of the Control Register File is mapped into this address space according to the setting of **Register Address Port**.

When the contents of the Register Address Port is one (RAP = 1) this block window is mapped to itself.
All read values are ZERO and write cycles have no effect.

3.3.2.22 MAC-Address Registers Link and maintenance

Address: 0x0100 Width [bit]: 6 x 8 (Link)

Address: 0x0110 Width [bit]: 6 x 8 (maintenance)

These registers hold the MAC Address.

MAC Addr_x: x maybe 1 or 3 for the two implemented link (1) and maintenance (3).

They are loaded at POWER ON RESET from the SPI Flash Memory or TWSI EEPROM.

They may be written in testmode.

Bit	Name	Description	Write	Read	Reset (HW)
MAC-Address Registers					
31:16	**Reserved**				
15:8	**MAC Addr_x<5>**	Mac-Address, Byte 5	ne	value	0
7:0	**MAC Addr_x<4>**	Mac-Address, Byte 4	ne	value	0
31:24	**MAC Addr_x<3>**	Mac-Address, Byte 3	ne	value	0
23:16	**MAC Addr_x<2>**	Mac-Address, Byte 2	ne	value	0
15:8	**MAC Addr_x<1>**	Mac-Address, Byte 1	ne	value	0
7:0	**MAC Addr_x<0>**	Mac-Address, Byte 0	ne	value	0

3.3.2.23 Interface Type Register (PMD Type and Connector Type)

Address: 0x0118 Width [bit]: 16

This register holds the type of interface.

It is loaded at POWER ON RESET from the SPI Flash Memory or the TWSI EEPROM.

It may be written in testmode.

Bit	Name	Description	Write	Read	Reset (HW)
Interface Type Register					
15:8	**PMD**	PMD Type	ne	value	0
7:0	**Connector**	Connector Type	ne	value	0

3.3.2.24 Chip Revision Register

Address: 0x011a Width [bit]: 8

This register holds chip revision number.

The Chip Revision is fixed to the current chip revision number.

Bit	Name	Description	Write	Read	Reset (HW)
Chip Revision Register					
7:4	**Chip Revision**	Initial Revision is 0x0 Fixed Value (incremented with chip revision)	ne	00	revision
3:0	**Reserved**		ne	value	value

3.3.2.25 Chip ID Register

Address: 0x011b Width [bit]: 8

This register holds the chip identification code.

Bit	Name	Description	Write	Read	Reset (HW)
Chip ID Register					
7:0	**Chip ID**	Initial ID is 0xb6 Fixed Value	ne	0xb6	

3.3.2.26 SPI Flash Memory Registers

Address: 0x011c Width [bit]: 4 x 8

These registers hold optional information.

They are loaded at POWER ON RESET from the SPI Flash Memory.

They are loaded at POWER ON RESET from the SPI Flash Memory.

They may be written in testmode (besides RAM Size field and bond status bits).

Bit	Name	Description	Write	Read	Reset (HW)
EPROM Registers					
31:24	**Eprom<3>**	EPROM, Byte 3	ne	value	0
Eprom<2> HW Resources					
23:17	**Reserved**		ne	value	0
16	**Link available**	0: Link not available (not implemented in device) 1: Link available	ne	value	0
Eprom<1> Clock Gating					
15:12	**Reserved**		ne	value	value
11	**bond_status Link**	Status of input signal (pin strapping): 0: Link active 1: Link inactive	ne	value	value
10	**Disable PHY/MAC Link**	Clock gating for ref_clk/mac_clk for Link 0: clock enabled for Link 1: clock disabled	ne	value	0
9	**Disable core_clk Link**	Clock gating for core_clk for Link 0: clock enabled for Link 1: clock disabled	ne	value	0
8	**Disable pci_clk Link**	Clock gating for pci_clk for Link 0: clock enabled for Link 1: clock disabled	ne	value	0
Eprom<0> RAM Size					
7:0	**Eprom<0> RAM size**	EPROM, Byte 0 RAM size specification: 0x0c: 12 x 4 KB = 48 KB (fixed value)	ne	0x0c	

3.3.2.27 Clock Divider Register

Address: 0x0120 Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
	Clock Division value		yes	aw	0
31:24					
23:16	**Clock Div Value**	Selection of divisor for clock division of core clock: 0: divide clock by 2 1: divide clock by 4 : n: divide clock by (n+1)*2 : 255: divide clock by 512 Minimum core clock frequency is 0.241 MHz (125 MHz/512)	yes	value	3
	Clock Division Control				
15:2	**Reserved**				
1	**Clock Div Enable**	Enable/Disable Clock division of core clock for power saving purposes Default: disabled 0b01	exec	0b10	0
0	**Clock Div Disable**			0b01	1

3.3.2.28 IRQ Timer Registers

Address: 0x0130 Width [bit]: 3 x 32

Usage of the **Timer** is described in chapter 3.23 Timer on page 278.

Bit	Name	Description	Write	Read	Reset (SW)
31:0	**Timer Init Value**		yes	aw	0
31:0	**Timer**		yes (for tests only)	value	0

CONFIDENTIAL
Document Classification: Proprietary Information

Bit	Name	Description	Write	Read	Reset (SW)
	Timer Control/Test				
31:16	**Reserved**				
		Test			
15:11	**Reserved**				
10	**Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Timer Test Off**			0b01	1
8	**Timer Step**	Timer decrement	exec	0	
		Control			
7:3	**Reserved**				
2	**Timer Start**	Start/Stop Timer	exec	0b10	0
1	**Timer Stop**			0b01	1
0	**Timer Clear IRQ**	Clear Timer Interrupt	exec	0	

The timer implements write posting and retries the following accesses to the timer while a posted write is in progress.

Target reads are retried until the addressed register is synchronized to PCI Clock.

3.3.2.29 IRQ Moderation Timer Registers

Address: 0x0140 Width [bit]: 3 x 32

Usage of the **IRQ Moderation Timer** is described in chapter 3.13 Interrupts on page 223.

Bit	Name	Description	Write	Read	Reset (SW)
31:0	**IRQ Moderation Timer Init Value**		yes	aw	0
31:0	**IRQ Moderation Timer**		yes	value	0
	IRQ Moderation Timer Control/Test				
31:16	**Reserved**				
		Test			
15:11	**Reserved**				
10	**IM Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**IM Timer Test Off**			0b01	1

Bit	Name	Description	Write	Read	Reset (SW)
8	**IM Timer Step**	Timer decrement	exec	0	
		Control			
7:3	**Reserved**				
2	**IM Timer Start**	Start/Stop IM Timer	exec	0b10	0
1	**IM Timer Stop**			0b01	1
0	**Reserved**				

The IRQ Moderation Timer implements write posting and retries the following accesses to the IRQ Moderation Timer while a posted write is in progress.

Target reads are retried until the addressed register is synchronized to PCI Clock.

3.3.2.30 Interrupt Moderation Mask Register

Address: 0x014c Width [bit]: 32

Each bit position defines, if the dedicated interrupt is moderated or propagated directly to the Interrupt Line **INTA#**.

The enable bits have the same bit positions as in the **Interrupt Source Register.** Unused bits are treated like reserved.

If set to ONE, interrupt is moderated.

If set to ZERO, interrupt is propagated directly to the Interrupt Line **INTA#.**

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Moderation Mask Register				
31:0	**En Mod IRQ xxx**	Enable Moderation Interrupt xxx	yes	aw	0

The interrupts of the **MAC Interrupt Source Register** cannot be moderated separately.

3.3.2.31 Interrupt Hardware Error Moderation Mask Register

Address: 0x0150 Width [bit]: 32

Each bit position defines, if the dedicated interrupt is moderated or propagated directly to the Interrupt Line **INTA#**.

The enable bits have the same bit positions as in the **Interrupt HW Error Source Register.** Unused bits are treated like reserved.

If set to ONE, interrupt is moderated.

If set to ZERO, interrupt is propagated directly to the Interrupt Line **INTA#.**

Bit	Name	Description	Write	Read	Reset (SW)
	Interrupt Hardware Error Moderation Mask Register				
31:0	**En Mod IRQ xxx**	Enable Moderation Hardware Interrupt xxx	yes	aw	0

3.3.2.32 General Purpose IO Register

Address: 0x015c Width [bit]: 32

For further extension the General Purpose IO Pins are routed to the General Purpose Registers. These IOs are programmable as inputs or outputs

Bit	Name	Description	Write	Read	Reset value
31	**Clock Debug Enable**	0 = Disable clock debug 1 = Enable clock debug Selected clock will be muxed out to VPD_CLK pin.	yes	aw	0
30	**Reserved**				
29:26	**CLK Debug**	0000 = PCI_CLK 0001 = CLK_PCI 0010 = CLK25 0111 = SPI_CLK 1001 = CLK_CORE 1010 = SMCLK_IN 1100 = PIPE_CLK	yes	aw	0
25:16	**GPIO Dir<9:0>**	Defines the type of the General Purpose IO Pins. 1 = output 0 = input In this application only GPIO<3:0>are available as external pins	yes	aw	0

Bit	Name	Description	Write	Read	Reset value
15:11	**Reserved**				
10	**Random Number**	Random Number Serializer Enable			
9	**Reserved**	Reserved			
8:0	**GPIO<8:0>**	These bits are routed to the ASIC's pins for future external options. As output they are synchronous to PCI Clock, as input they are synchronized to PCI Clock. In this application only GPIO<3:0>are available as external pins	yes	aw	0

3.3.2.33 TWSI (HW) Registers

Address: 0x0160 Width [bit]: 3 x 32

Bit	Name	Description	Write	Read	Reset value
	TWSI (HW) Control Register (Address: 0x0160, width 32)				
31	**Flag**	Starts the TWSI data transfers, determines its direction and signals its completion by being toggled by HW. If written 1, a TWSI write is started. Set to 0 after completion. If written 0, a TWSI read is started. Set to 1 after completion. Generates an interrupt on completion.	exec	value	0
30:16	**TWSI Address**	Address of the TWSI device register to be written/read.	yes	aw	0x00
15:9	**TWSI Devsel**	Devsel Byte of the TWSI device to be written/read.	yes	aw	0x00
8:5	**Reserved**				
4	**TWSI Burst**	0 = single Byte transfers 1 = 4 Byte Page Mode write transfers with fixed page size of 8 Bytes assumed	yes	aw	0

Bit	Name	Description	Write	Read	Reset value
3:1	**TWSI Device Size**	Defines the size of the addressed TWSI Device in Bytes. 0: 256 Bytes and smaller 1: 512 Bytes 2: 1024 Bytes 3: 2048 Bytes 4: 4096 Bytes 5: 8192 Bytes 6: 16384 Bytes 7: 32768 Bytes Default value is 256 Bytes.	yes	aw	0x00
0	**TWSI Stop**	A written 1 interrupts the current TWSI transfer at the next byte boundary with a stop condition and signals end of TWSI transfer by toggling **Flag**.	exec	0	0
	TWSI (HW) Data Register (Address: 0x0164, width 32)				
31:0	**TWSI Data**	Must be written before TWSI Address Register for TWSI write. Contains TWSI read Data after completion of TWSI read.	yes	aw	0x00
	TWSI (HW) IRQ Register (Address: 0x0168, width 32)				
31:1	**Reserved**				
0	**Clear IRQ TWSI**	Clears Interrupt Request from TWSI HW interface	exec	aw	0

These registers implement a serial TWSI interface to the optional temperature/voltage sensor. HW runs the 100 kHz serial TWSI protocol to obtain data.

The HW controlled TWSI interface and the SW controlled TWSI interface are connected to the same TWSI bus (Pins TWSI_DATA and TWSI_Clock). They **MUST NOT** be used in parallel.

If the HW controlled TWSI interface is used, the **TWSI (SW) Register** has to be set to inactive values (Reset values).

If the SW controlled TWSI interface is used, the HW controlled TWSI interface MUST NOT be started (**FLAG/ TWSI (HW) Register).**

The TWSI clock and data port pins are pulled high by a pull up resistor to VCC of the TWSI device.

3.3.2.34 TWSI (SW) Register

Address: 0x016c Width [bit]: 32

Bit	Name	Description	Write	Read	Reset (SW)
TWSI (SW) Register					
31:3	**Reserved**				
2	**TWSI Data Dir**	Defines direction of TWSI Data Port: 0 = Input 1 = Output	yes	aw	0
1	**TWSI Data**	TWSI Interface Data Port	yes	dir=0: value dir=1: aw	0
0	**TWSI Clock**	TWSI Interface Clock	yes	aw	1

This register implements a serial TWSI interface to the optional temperature/voltage sensor. SW has to run the serial TWSI protocol to obtain data.

As output, the **Data Port** must simulate an open collector output in order to obtain a 0.7 Vcc signal level at the TWSI device (if supplied with 5 V).

Driving to low level: **TWSI Data =** 0

TWSI Data Dir = 1

Floating to high level:**TWSI Data =** x

TWSI Data Dir = 0

The HW controlled TWSI interface and the SW controlled TWSI interface are connected to the same TWSI bus (Pins TWSI_DATA and TWSI_Clock). They **MUST NOT** be used in parallel.

If the HW controlled TWSI interface is used, the **TWSI (SW) Register** has to be set to inactive values (Reset values).

If the SW controlled TWSI interface is used, the HW controlled TWSI interface MUST NOT be started (**FLAG/ TWSI (HW) Register).**

The TWSI clock and data port pins are pulled high by a pull up resistor to VCC of the TWSI device.

3.3.2.35 PEX PHY Address/Data Registers

Address: 0x0170 Width [bit]: 32

The PHYs of the PCI Express part are made accessible via the following registers.

Bit	Name	Description	Write	Read	Reset (SW)
PEX PHY Address Register					
31	**PEX PHY Access Mode**	0: Write access to PEX PHY register 1: Read PEX PHY register and store data into data region bits 15:0 of this register.	yes	aw	0
30	**PEX PHY regfile**	0: PHY's register file of PEX module 1: PEX debug register file	yes	aw	0
29:16	**PEX PHY Address**	14 bit PHY/debug Address for PCI Express module	yes	aw	0
PEX PHY Data Register					
15:0	**PEX PHY Data**	16 bit PHY Data for PCI Express Module	yes	aw	0

3.3.2.36 RAM Random Registers

Address: 0x0180 - 0x0188 Width [bit]: 3 x 32

Bit	Name	Description	Write	Read	Reset (SW)
RAM Address (offset: 0x00, width: 32)					
31:19	**Reserved**				
18:0	**RAM Address**	Defines RAM address in qwords.	yes	aw	0
Data Port/lower dword (offset: 0x04, width: 32)					
31:0	**Data Port/lower dword**	Dataport/lower dword for exchange of read/write data. On a PCI read access, reading from RAM to the data ports is initiated. The initiating PCI access and subsequent accesses are retried on PCI while reading from RAM	yes	value exec	0

Bit	Name	Description	Write	Read	Reset (SW)
Data Port/upper dword (offset: 0x08, width: 32)					
31:0	**Data Port/upper dword**	Dataport/upper dword for exchange of read/write data. On a PCI write access, writing to RAM from the data ports is initiated. Subsequent PCI accesses are retried on PCI while writing to RAM	yes exec	value	0

3.3.2.37 RAM Interface Registers

Address: 0x0190 - 0x01a0 Width [bit]: 5 x 32

The Timeout values limit the burst length of data transfers for each requestor individually.

Default values must be used.

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Timeout Init Values 0 - 3 (offset: 0x00, width: 32)				
31:24	**Timeout Init Value 3**	Read SM Rx1	yes	aw	32
23:16	**Reserved**	Write SM Txs1	yes	aw	32
15:8	**Timeout Init Value 1**	Write SM Txa	yes	aw	32
7:0	**Timeout Init Value 0**	Write SM Rx	yes	aw	32
	Timeout Init Values 4 and 5 (offset: 0x04, width: 32)				
31:16	**Reserved (leg-acy)**				
15:8	**Reserved**		yes	aw	32
7:0	**Timeout Init Value 4**	Read SM Txa	yes	aw	32
31:0	**Reserved**				

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Timeout Timer (offset: 0x0c, width: 32)				
31:8	**Reserved**				
7:0	**Timeout Timer**		yes (for tests only)	value	0
	RAM Interface Control/Test (offset: 0x10, width: 32)				
		Test			
31:20	**Reserved**				
19	**Timeout**	1: if Timeout Timer = 0	ne	value	1
18	**Timeout Timer Test On**	Testmode ON/OFF	exec	0b10	0
17	**Timeout Timer Test Off**			0b01	1
16	**Timeout Timer Step**	Timer decrement	exec	0	
		Control			
15:10	**Reserved**				
9	**Clear IRQ PAR Rd RAM**	Clear Parity Error on read Interrupt	exec	0	
8	**Clear IRQ PAR Wr RAM**	Clear Parity Error on write Interrupt	exec	0	
7:2	**Reserved**				
1	**Reset Clear**	Set/Clear **Reset.** Executed, if appropriate bit is set to 1. If **Reset** is set, all RAM Interface functions and registers are in their reset state. **Reset** is forwarded to the RAM Random Access Device in order to avoid a hangup on a target access while the RAM Interface is reset.	exec	0b10	0
0	**Reset Set**			0b01	1 (SW)

3.3.2.38 Transmit Arbiter Registers

Address: 0x0200 - 0x0210 Width [bit]: 5 x 32

Bit	Name	Description	Write	Read	Reset (SW)
	Interval Timer Init Value (offset: 0x00, width: 32)				
31:0	**Interval Timer Init Value**	Number of core clock cycles as time interval for rate control. T_{Max} = 27.53 s	yes	aw	0
	Interval Timer (offset: 0x04, width: 32)				
31:0	**Interval Timer**	Interval Timer value: number of core clock cycles (125 MHz) T_{Max} = 27.53 s	yes	aw	0
	Limit Counter Init Value (offset: 0x08, width: 32)				
31:24	**Reserved**				
23:0	**Reserved**		yes	aw	0
	Limit Counter (offset: 0x0c, width: 32)				
31:0	**Limit Counter**		yes	aw	0
	Timer/Counter Control/Status/Test (offset: 0x10, width: 32)				
		Status			
31:17	**Reserved**				
16	**Priority/Sync. Rambuffer**	Set as long Limit Counter not ZERO or if **Force Sync On** is set	ne	value	0
		Test			
15:14	**Reserved**				
13	**Interval Timer Test On**	Testmode ON/OFF	exec	0b10	0
12	**Interval Timer Test Off**			0b01	1
11	**Interval Timer Step**	Timer decrement	exec	0	

Bit	Name	Description	Write	Read	Reset (SW)
10	**Limit Counter Test On**	Testmode ON/OFF	exec	0b10	0
9	**Limit Counter Test Off**			0b01	1
8	**Limit Counter Step**	Counter decrement	exec	0	
		Control			
6	**Force Sync Off**			0b01	1
4	**Dis Alloc**			0b01	1
2	**Rate Ctrl Stop**			0b01	1
1	**Arbiter Operational On**	Operational mode ON/OFF	exec	0b10	0
0	**Arbiter Operational Off**	If OFF, no request is granted. Has to be set to OFF until all other devices are initialized		0b01	1

3.3.2.39 RSS Key Registers

Address: 0x0220 - 0x022c Width [bit]: 4 x 32

Address: 0x02a0 - 0x02ac Width [bit]: 4 x 32

RSS is described in “Receive-Side Scaling Hash Design Specification” by Microsoft Corporation.

Bit	Name	Description	Write	Read	Reset (SW)
RSS Key 0 (offset: 0x00, width: 32)					
31:0	**RSS Key 0**	Part 0 (LSB) of random key for RSS Hash Algorithm	yes	aw	0
RSS Key 1 (offset: 0x04, width: 32)					
31:0	**RSS Key 1**	Part 1 of random key for RSS Hash Algorithm	yes	aw	0

Bit	Name	Description	Write	Read	Reset (SW)
RSS Key 2 (offset: 0x08, width: 32)					
31:0	**RSS Key 2**	Part 2 of random key for RSS Hash Algorithm	yes	aw	0
RSS Key 3 (offset: 0x0c, width: 32)					
31:0	**RSS Key 3**	Part 3 (MSB) of random key for RSS Hash Algorithm	yes	aw	0

3.3.2.40 PCI Configuration Registers

Address: 0x0380 - 0x03fc Width [bit]: 32 x 32

The Configuration Register File is mapped additionally into the Control Register File.

At this place the lower 128 bytes of the Configuration Register Files are mapped. The whole Configuration Register File is mapped to blocks 56 up to 60.

It is read only with some exceptions: **Our Register 1** and **2** and **VPD Address** and **VPD Data Register** may be written.

Write operations are completed normally on the bus and the data is discarded.

For testing purposes, the Configuration Register File may be set writable by **En Config Write**.

3.3.2.41 BMU Registers for Receive Queues

Address: 0x0400 - 0x044c Width [bit]: 20 x 32 (receive queue , structure as shown below)

These registers are intended to be used for testing and diagnostic purposes, except the **Control/Status Registers** and **Watermark** for initialization.

Manipulating the content of these registers under ‘normal’ running conditions is not recommended and may lead to undefined results.

If the default values are acceptable, the **Watermark** should not be written (for future backward compatibility, if watermarks/FIFO depth are adapted).

For each receive queue in each link a separate set of registers is implemented.

Bit	**Name**	**Description**	Write	Read	Reset (SW)
Receive Queue Registers					
	Current Receive Descriptor (offset: 0x00, width: 2 x 16)				
31:16	**Receive Buffer Control**	Highest 16 bit of current List Element (when of type buffer) Own bit, Opcode, Control bits	TO	value	0
15:0	**Receive Buffer Byte Count**		TO	value	0
	RSS Hash Checksum (offset: 0x04, width: 32)				
31:0	**RSS Hash Check-sum**	Result of RSS Hash calculation, when enabled.			
	Receive Buffer Address (offset: 0x08, width: 2 x 32)				
31:0	**Receive Buffer Address Lo**	Lower part of receive buffer address	TO	value	0
31:0	**Receive Buffer Address Hi**	Higher part of receive buffer address	TO	value	0
	Receive Buffer Status Word (offset: 0x10, width: 32)				
31:0	**RFSW**	Receive Frame Status Word as defined by GMAC (including length). Valid, if **Own** is cleared by the BMU and **EOF** is set. If not valid, the BMU writes back, what received at that place.	TO	value	0
	Receive Timestamp (offset: 0x14, width: 32)				
31:0	**Timestamp**	Receive Timestamp as defined by the **Timestamp Timer** at writing the Receive Status Word to the Receive MAC FIFO. Valid, if **Own** is cleared by the BMU and **EOF** is set. If not valid, the BMU writes back, what received at that place.	TO	value	0

Bit	Name	Description	Write	Read	Reset (SW)
	TCP/IP Checksum Extension (offset: 0x18, width: 4 x 16)				
	The list element TCPPAR containing the start addresses of the two checksums enables TCP/IP checksum calculation. TCP/IP checksums 1 & 2 (16 bit) are calculated from TCP Sum Start 1 & 2 up to the end of the packet. The checksums are written to TCP Sum 1 & 2. The byte order within the TCP sum is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8). Note: An FCS appended to the frame by the GMAC is also added to the checksum. If the calculation does not end 16-bit-aligned at the end of the packet, the missing byte is seen as 0. Registers holding these values are updated only, if **STF** is set (except **TCP Sum 1 & 2**)				
31:16	**Checksum 2**	Checksum 2 Valid, if **Own** is cleared by the BMU and **EOF** is set. Defined by the PCI device, if EOF is set	TO	value	0
15:0	**Checksum 1**	Checksum 1 Valid, if **Own** is cleared by the BMU and **EOF** is set. Defined by the PCI device, if EOF is set	TO	value	0
31:16	**Start position 2**	Checksum 2, start position for calculation in bytes (16-bit aligned) counted from zero (first byte = 0) Defined by the host, if STF is set	TO	value	0
15:0	**Start position 1**	Checksum 1, start position for calculation in bytes (16-bit aligned) counted from zero (first byte = 0) Defined by the host, if STF is set	TO	value	0
	VLAN Tag (offset: 0x20, width: 16)				
31:16	**Reserved**		TO	value	0
15:0	**VLAN Tag**	received VLAN Tag bytes out of data stream The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).			

Bit	Name	Description	Write	Read	Reset (SW)
	Done Index (offset: 0x24, width: 16)				
31:12	**Reserved**				
11:0	**Done Index**	Incremented for each processed list element	TO	value	0
	Request Address (offset: 0x28, width: 2 x 32)				
31:0	**Request Addr Lo**	Lower 32 bit of current request address	TO	value	0
31:0	**Request Addr Hi**	Higher 32 bit of current request address	TO	value	0
	Request Byte Count (offset: 0x30, width: 16)				
31:11	**Reserved**				
10:0	**Request Byte Count**	Length of current request in number of bytes	TO	value	0
	BMU Control/Status Register (offset: 0x34, width: 32)				
31	**BMU Idle**	Status bit BMU in Idle state	ne	value	1
30	**TCP Pkt**	Flag: ‘1’, if current packet is TCP/IP Only valid, when RSS Hash enabled	TO	value	0
29	**IP Pkt**	Flag: ‘1’, if current packet is IP Only valid, when RSS Hash enabled	TO	value	0
28:16	**Reserved**				
15	**RSS Hash Enable**	Enables/disables RSS hash calculation for receive queue Default: Disabled (0b01)	exec	0b10	0
14	**RSS Hash Disable**			0b01	1
13	**RX Checksum Enable**	Enables/disables TCP/IP checksum check Default: Disabled (0b01)	exec	0b10	0
12	**RX Checksum Disable**			0b01	1
11	**Clear IRQ Parity**	Clear IRQ on Parity errors (RX BMU FIFO read data parity error)	exec	0	0
10	**Clear IRQ Check**	Clear IRQ Check. (This interrupt is asserted when receiving a wrong opcode within a list element)	exec	0	0

Bit	Name	Description	Write	Read	Reset (SW)
9	**Stop Rx BMU**	10: Stop receive queue transfer after next end of packet 01: normal operation (default)	exec	0b10	0
8	**Start Rx BMU**			0b01	1
7	**FIFO Operational On**	Default: OFF (0b01) Must be switched to ON after initialization	exec	0b10	0
6	**FIFO Operational Off**			0b01	1
5	**FIFO Enable**	Default: Reset (0b01)	exec	0b10	0
4	**FIFO Reset**			0b01	1
3	**BMU Operational On**	Default: OFF (0b01) Must be switched to ON after initialization	exec	0b10	0
2	**BMU Operational Off**			0b01	1
1	**BMU Enable**	Default: Reset (0b01)	exec	0b10	0
0	**BMU Reset**			0b01	1
	BMU Test Register (offset: 0x38, width: 32)				
31:23	**Reserved**				
22	**Testmode Shadow Read Ptr On**	Switch to testmode Shadow Read Pointer Default: operation (testmode = off)	exec	0b10	0
21	**Testmode Shadow Read Ptr Off**			0b01	1
20	**Teststep Shadow Read Ptr**	Teststep: increment Shadow Read Pointer	exec	0	0
19	**Reserved**				
18	**Testmode Read Ptr On**	Switch to testmode Read Pointer Default: operation (testmode = off)	exec	0b10	0
17	**Testmode Read Ptr Off**			0b01	1
16	**Teststep Read Ptr**	Teststep: increment Read Pointer	exec	0	0

Bit	**Name**	**Description**	Write	Read	Reset (SW)
15	**Reserved**				
14	**Testmode Shadow Write Ptr On**	Switch to testmode Shadow Write Pointer Default: operation (testmode = off)	exec	0b10	0
13	**Testmode Shadow Write Ptr Off**			0b01	1
12	**Teststep Shadow Write Ptr**	Teststep: increment Shadow Write Pointer	exec	0	0
11	**Reserved**				
10	**Testmode Write Ptr On**	Switch to testmode Write Pointer Default: operation (testmode = off)	exec	0b10	0
9	**Testmode Write Ptr Off**			0b01	1
8	**Teststep Write Ptr**	Teststep: increment Write Pointer	exec	0	0
7	**Reserved**				
6	**Testmode Req Nbytes/Addr On**	Switch to testmode Request Nbytes/Addresses Default: operation (testmode = off)	exec	0b10	0
5	**Testmode Req Nbytes/Addr Off**			0b01	1
4	**Teststep Req Nbytes/Addr**	Teststep Request Nbytes/Addresses An acknowledge from BIU is emulated and as a result address counter is incremented by Nbytes and Nbytes is reset to zero.	exec	0	0
3	**Reserved**				
2	**Testmode Done Index On**	Switch to testmode Done Index Default: operation (testmode = off)	exec	0b10	0
1	**Testmode Done Index Off**			0b01	1
0	**Teststep Done Index**	Teststep: increment Done Index	exec	0	0
	BMU Statemachine Register (offset: 0x3c, width: 32)				
31:0		State variables of all statemachines (writable only, when in operational mode, but then remains not the written value)	TO	value	0x00 (idle state)

Bit	Name	Description	Write	Read	Reset (SW)
	FIFO Watermark (offset: 0x40, width: 16)				
15:11	**Reserved**				
10:0		FIFO Watermark (bytes)	yes tbd: ITO?	aw	0x600
	FIFO Alignment (offset: 0x42, width: 8)				
15:7	**Reserved**				
6:4	**MUX**	Multiplexer setting	ne	value	0
3	**Reserved**				
2:0	**VRAM**	VRAM position	yes	value	0
	FIFO Read Shadow Pointer (offset: 0x44, width: 16)				
15:11	**Reserved**				
10:0		FIFO Read Shadow Pointer (bytes)	TO	value	0
	FIFO Read Shadow Level (offset: 0x46, width: 8)				
15:8	**Reserved**				
7:0		FIFO Read Shadow Level (qword)	ne	value	0
	FIFO Read Pointer (offset: 0x48, width: 8)				
15:8	**Reserved**				
7:0		FIFO Read Pointer (qword)	TO	value	0
	FIFO read Level (offset: 0x4a, width: 8)				
15:8	**Reserved**				
7:0		FIFO read Level (qword)	ne	value	0
	FIFO Write Pointer (offset: 0x4c, width: 8)				
7:0		FIFO Write Pointer (qword)	TO	value	0
	FIFO Write Shadow Pointer (offset: 0x4d, width: 8)				
7:0		FIFO Write Shadow Pointer (qword)	TO	value	0
	FIFO Write Level (offset: 0x4e, width: 8)				
7:0		FIFO Write Level (qword)	ne	value	0

Bit	Name	Description	Write	Read	Reset (SW)
	FIFO Write Shadow Level (offset: 0x4f, width: 8)				
7:0		FIFO Write Shadow Level (qword)	ne	value	0

3.3.2.42 BMU Registers for Transmit Queues

Address: 0x0680 - 0x06cc Width [bit]: 20 x 32 (asynchronous transmit queue)

These registers are intended to be used for testing and diagnostic purposes, except the **Control/Status Registers** and **Watermark** for initialization.

Manipulating the contents of these registers under 'normal' running conditions is not recommended and may lead to undefined results.

If the default values are acceptable, the **Watermark** should not be written (for future backward compatibility, if watermarks/FIFO depth are adapted).

The **Watermark** of the transmit queues **MUST NOT BE SET** to zero.

For each transmit queue in each link a separate set of registers is implemented.

The following register set is implemented for asynchronous transmit queue.

Bit	Name	Description	Write	Read	Reset (SW)
Transmit Queue Registers					
	Current Transmit Descriptor (offset: 0x00, width: 2 x 16)				
31:16	**Transmit Buffer Control**		TO	value	0
15:0	**Transmit Buffer Byte Count**		TO	value	0
	Transmit Buffer Address (offset: 0x08, width: 2 x 32)				
31:0	**Transmit Buffer Address Lo**	Lower part of transmit buffer address	TO	value	0
31:0	**Transmit Buffer Address Hi**	Higher part of transmit buffer address	TO	value	0

Bit	Name	Description	Write	Read	Reset (SW)
	Transmit Buffer Status Word (offset: 0x10, width: 32)				
31:0	**TFSW**	Transmit Frame Status Word as defined by the MAC. Appended to transmit data of this buffer, if **EOF** is set. This is a placeholder, because GMAC does not expect a Transmit Frame Status Word. Default value is 0. In loopback mode Transmit Frame Status Word has to be set to the expected Receive Frame Status Word of GMAC. Defined by the host, if EOF is set	TO	value	0
	TCP/IP Checksum Extension (offset: 0x18, width: 4 x 16)				
31:16	**Reserved**				
15:0	**TCP Sum Init**	Checksum, start value Defined by the host, if STF is set. The byte order within the TCP sum init value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8). During TCP Sum calculation the MSB is added to the first, third and so on byte of the packet. The LSB is added to the second, forth and so on byte of the packet.	TO	value	0
31:16	**TCP Sum Start**	Checksum, start position for calculation in bytes (16-bit-aligned) counted from zero (first byte = 0) Defined by the host, if STF is set.	TO	value	0
15:0	**TCP Sum Write**	Checksum, write position for checksum in bytes (16-bit-aligned) counted from zero (first byte = 0) Defined by the host, if STF is set.	TO	value	0

Bit	**Name**	**Description**	Write	Read	Reset (SW)
	VLAN Tag (offset: 0x20, width: 16)				
31:16	**Reserved**				
15:0	**VLAN Tag**	Transmit VLAN Tag bytes extracted from received descriptor of host buffer The byte order within the VLAN Tag value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).	TO	value	0
	Done Index (offset: 0x24, width: 16)				
31:12	**Reserved**				
11:0	**Done Index**	Incremented for each processed list element	TO	value	0
	Request Address (offset: 0x28, width: 2 x 32)				
31:0	**Request Addr Lo**	Lower 32 bit of current request address	TO	value	0
31:0	**Request Addr Hi**	Higher 32 bit of current request address	TO	value	0
	Request Byte Count (offset: 0x30, width: 16)				
31:11	**Reserved**				
10:0	**Request Byte Count**	Length of current request in number of bytes	TO	value	0
	BMU Control/Status Register (offset: 0x34, width: 32)				
31	**BMU Idle**	Status bit BMU in Idle state	ne	value	1
30:14	**Reserved**				
13	**IP ID Increment Enable**	Enables/disables incrementing of the IP identification field with every segment during TCP segmentation. Default: disabled (0b01)	exec	0b10	0
12	**IP ID Increment Disable**			0b01	1
11	**Clear IRQ TCP**	Clear IRQ on TCP segmentation length mismatch	exec	0	0
10	**Clear IRQ Check**	Clear IRQ Check. (This interrupt is asserted when receiving a wrong opcode within a list element)	exec	0	0
9	**Stop Tx BMU**	10: Stop transmit queue transfer after next end of packet 01: normal operation (default)	exec	0b10	0
8	**Start Tx BMU**			0b01	1

Bit	Name	Description	Write	Read	Reset (SW)
7	**FIFO Operational On**	Default: OFF (0b01) Must be switched to ON after initialization	exec	0b10	0
6	**FIFO Operational Off**			0b01	1
5	**FIFO Enable**	Default: Reset (0b01)	exec	0b10	0
4	**FIFO Reset**			0b01	1
3	**BMU Operational On**	Default: OFF (0b01) Must be switched to ON after initialization	exec	0b10	0
2	**BMU Operational Off**			0b01	1
1	**BMU Enable**	Default: Reset (0b01)	exec	0b10	0
0	**BMU Reset**			0b01	1
	BMU Test Register (offset: 0x38, width: 32)				
31:23	**Reserved**				
22	**Testmode Shadow Read Ptr On**	Switch to testmode Shadow Read Pointer Default: operation (testmode = off)	exec	0b10	0
21	**Testmode Shadow Read Ptr Off**			0b01	1
20	**Teststep Shadow Read Ptr**	Teststep: increment Shadow Read Pointer	exec	0	0
19	**Reserved**				
18	**Testmode Read Ptr On**	Switch to testmode Read Pointer Default: operation (testmode = off)	exec	0b10	0
17	**Testmode Read Ptr Off**			0b01	1
16	**Teststep Read Ptr**	Teststep: increment Read Pointer	exec	0	0
15	**Reserved**				
14	**Testmode Shadow Write Ptr On**	Switch to testmode Shadow Write Pointer Default: operation (testmode = off)	exec	0b10	0
13	**Testmode Shadow Write Ptr Off**			0b01	1
12	**Teststep Shadow Write Ptr**	Teststep: increment Shadow Write Pointer	exec	0	0

Bit	**Name**	**Description**	Write	Read	Reset (SW)
11	**Reserved**				
10	**Testmode Write Ptr On**	Switch to testmode Write Pointer Default: operation (testmode = off)	exec	0b10	0
9	**Testmode Write Ptr Off**			0b01	1
8	**Teststep Write Ptr**	Teststep: increment Write Pointer	exec	0	0
7	**Reserved**				
6	**Testmode Req Nbytes/Addr On**	Switch to testmode Request Nbytes/Addresses Default: operation (testmode = off)	exec	0b10	0
5	**Testmode Req Nbytes/Addr Off**			0b01	1
4	**Teststep Req Nbytes/Addr**	Teststep Request Nbytes/Addresses An acknowledge from BIU is emulated and as a result address counter is incremented by Nbytes and Nbytes is reset to zero.	exec	0	0
3	**Reserved**				
2	**Testmode Done Index On**	Switch to testmode Done Index Default: operation (testmode = off)	exec	0b10	0
1	**Testmode Done Index Off**			0b01	1
0	**Teststep Done Index**	Teststep: increment Done Index	exec	0	0
	BMU Statemachine Register (offset: 0x3c, width: 32)				
31:0		state variables of all statemachines	TO	value	0x00 (idle state)
	FIFO Watermark (offset: 0x40, width: 11)				
15:11	**Reserved**				
10:0		FIFO Watermark (bytes)	yes	aw	0x600
	FIFO Alignment (offset: 0x42, width: 8)				
15:7	**Reserved**				
6:4	**MUX**	Multiplexer setting	ne	value	0

Bit	Name	Description	Write	Read	Reset (SW)
3	**Reserved**				
2:0	**VRAM**	VRAM position	ne	value	0
	FIFO Write Shadow Pointer (offset: 0x44, width: 16)				
15:11	**Reserved**				
10:0		FIFO Write Shadow Pointer (bytes)	TO	value	0
	FIFO Write Shadow Level (offset: 0x46, width: 8)				
15:8	**Reserved**				
7:0		FIFO write Level (qword)	ne	value	0
	FIFO Write Pointer (offset: 0x48, width: 8)				
15:8	**Reserved**				
7:0		FIFO Write Pointer (qword)	TO	value	0
	FIFO Write Level (offset: 0x4a, width: 8)				
15:8	**Reserved**				
7:0		FIFO write Level (qword)	ne	value	0
	FIFO Read Pointer (offset: 0x4c, width: 8)				
15:8	**Reserved**				
7:0		FIFO Read Pointer (qword)	TO	value	0
	FIFO read Level (offset: 0x4e, width: 8)				
15:8	**Reserved**				
7:0		FIFO read Level (qword)	ne	value	0

3.3.2.43 Prefetch Unit Registers

Address: Base Address Width [bit]:6 x 32
Address: Base Address + 0x20 Width [bit]:4 x 8

Each transmit or receive queue has its dedicated Prefetch FIFO and the Prefetch Unit Registers.

Base Addresses for the Prefetch Unit Register sets:

queue	Base Address
RX	0x0450
TX	0x06d0

The register address of each register is calculated by adding the base address and the offset.

Due to a smaller PFU FIFO for transmit queues, the size of the pointers and levels is to be decreased by one for transmit PFUs.

Bit	**Name**	**Description**	**Write**	**Read**	**Reset (Link)**
	Prefetch Control Register (Offset: 0x00, Width: 32)				
31:15	**Reserved**				
14	**Master Request Loopback Test On**	Testmode ON/OFF for the Master request, i.e. enables the stepping of the Master request by the SW. The default operation is disabled.	exec	0b10	0
13	**Master Request Loopback Test Off**			0b01	1
12	**Master Request Loopback Step**	Enables the Loopback of a single master request. For execution of this command, the Master Request Loopback Test **MUST** be set to ON.	exec	0	0
11	**Reserved**				
10	**FIFO Read Pointer Test On**	Testmode ON/OFF for the FIFO Read Pointer, i.e. enables the stepping of the pointer by the SW. The default operation is disabled.	exec	0b10	0
9	**FIFO Read Pointer Test Off**			0b01	1

Bit	Name	Description	Write	Read	Reset (Link)
8	**FIFO Read Pointer Step**	FIFO Read Pointer increment. For execution of this command, the FIFO Read Pointer Test **MUST** be set to ON.	exec	0	0
7	**Reserved**				
6	**FIFO Write Pointer Test On**	Testmode ON/OFF for the FIFO Write Pointer, i.e. enables the stepping of the pointer by the SW. The default operation is disabled.	exec	0b10	0
5	**FIFO Write Pointer Test Off**			0b01	1
4	**FIFO Write Pointer Step**	FIFO Write Pointer increment. For execution of this command, the FIFO Write Pointer Test **MUST** be set to ON.	exec	0	0
3	**Operational On**	Operational mode ON/OFF OFF resets all activities of the prefetch unit for initialization of the pointers and registers. The default operation is disabled.	exec	0b10	0
2	**Operational Off**			0b01	1
1	**PFU Reset Clear**	Set/Clear **PFU Reset.** Executed, if appropriate bit is set to 1. If **PFU Reset** is set, the integrated PFU and the PFU Target Interface are in their reset state.	exec	0b10	0
0	**PFU Reset Set**			0b01	1
List Last Index (Offset: 0x04, Width: 12)					
31:12	**Reserved**				
11:0	**List Last Index**	Specifies the index of the last entry in the list. The descriptor element length is defined by List Last Index + 1. The length is given as number of list elements Bit 6:0: fixed value 0x7f	yes	aw	0x07f
List Start Address Low (Offset: 0x08, Width: 32)					
31:12		Identifies the lower 32 bit of the start address of the descriptor list in the system memory.	yes	aw	0x0
11:0		The address is always aligned to 4 KB boundaries, i.e. the lower bits are always 0x0.	ne	0x0	0x0

Bit	Name	Description	Write	Read	Reset (Link)
List Start Address High (Offset: 0x0c, Width: 32)					
31:0		Identifies the upper 32 bit of the start address of the descriptor list in the system memory.	yes	aw	0x0
Get Index (Offset: 0x10, Width: 32)					
31:12	**Reserved**				
11:0		Specifies the Pop/Rd address within the descriptor list in system memory. This Get Index points to the address of the next element to be fetched. If Get Index and Put Index are equal, no element is in the FIFO The index is given in list element granularity.	yes	value	0x0
Put Index (Offset: 0x14, Width: 12)					
31:12	**Reserved**				
11:0		Specifies the Push/Wr address within the descriptor list in system memory. This Put Index points to the address of the next free element in the descriptor list (i.e. the position after the last valid element in the descriptor list). If Get Index and Put Index are equal, no element is in the FIFO. The index is given in list element granularity.	yes	aw	0x0
FIFO RAM Write and Write Shadow Pointer (Offset: 0x20, Width: 32)					
31:24	**Reserved**				
23:16	**FIFO Wr Ptr**	Specifies the Push/Wr pointer to the internal prefetch FIFO RAM. The pointer is given in bytes.	yes	value	0x0
FIFO RAM Write Pointer					
15:5	**Reserved**				
4:0	**FIFO Rd Ptr**	Specifies the Push/Wr pointer to the internal prefetch FIFO RAM. The pointer is given in list element granularity (qwords).	yes	value	0x0

Bit	Name	Description	Write	Read	Reset (Link)
FIFO RAM Read Pointer (Offset: 0x24, Width: 6)					
31:5	**Reserved**				
4:0	**FIFO Rd Ptr**	Specifies the Pop/Rd pointer to the internal prefetch FIFO RAM. The pointer is given in list element granularity.	yes	value	0x0
Master Request nbytes (Offset: 0x28, Width: 32)					
31:27	**Reserved**				
26:16	**Master Request nBytes**	Specifies the number of bytes requested from the master. This value is used only for debugging purpose. The number is given in Bytes	no	value	0x0
FIFO Watermark					
15:6	**Reserved**				
5:0	**FIFO Watermark**	Specifies the watermark level (i.e. free elements level) for prefetching new descriptor elements. The watermark is given in list element granularity.	yes	aw	0x0

Bit	Name	Description	Write	Read	Reset (Link)
FIFO Shadow level (Offset: 0x2C, Width: 32)					
31:24	**Reserved**				
23:16	**FIFO Shadow level**	Specifies the current filling level of the prefetch FIFO RAM. This value is directly updated by the read and write accesses to the FIFO RAM. This value is used for debugging purpose only. The number is given in bytes.	yes	value	ram size
FIFO level					
15:5	**Reserved**				
4:0	**FIFO level**	Specifies the current filling level of the prefetch FIFO RAM. This value is directly updated by the read and write accesses to the FIFO RAM. This value is used for debugging purpose only. The number is given in list element granularity (qwords).	yes	value	ram size

3.3.2.44 Receive Rambuffer Registers

Address: 0x0800 - 0x0828 Width [bit]: 11 x 32 (receive queue, structure as shown below)

Initialization or re-arrangement of a Rambuffer should ever start from reset state.

Bit	Name	Description	Write	Read	Reset (private)
	Receive Rambuffer Start Address (Offset: 0x00, width: 32)				
31:13	**Reserved**				
12:0	**Start Address**	Start Address in qwords of this queue in internal memory. Bit [7:0] are replaced by ZEROes internally, which means that the address is limited to multiples of 1 kB. Has to be defined after each Reset.	yes (for init/ tests only)	aw	0

Bit	Name	Description	Write	Read	Reset (private)
	Receive Rambuffer End Address (Offset: 0x04, width: 32)				
31:13	**Reserved**				
12:0	**End Address**	End Address in qwords of this queue in internal memory. Bit [7:0] are replaced by ONEs internally, which means that the address is limited to multiples of 1 kB. Has to be defined after each Reset.	yes (for init/ tests only)	aw	0
	Receive Buffer Write Pointer (Offset: 0x08, width: 32)				
31:13	**Reserved**				
12:0	**Write Pointer**	Write Pointer in qwords. Has to be set to **Receive Rambuffer Start Address** after each Reset.	yes (for init/ tests only)	value	0
	Receive Rambuffer Read Pointer (Offset: 0x0c, width: 32)				
31:13	**Reserved**				
12:0	**Read Pointer**	Read Pointer in qwords Has to be set to **Receive Rambuffer Start Address** after each Reset.	yes (for init/ tests only)	value	0
	Receive Rambuffer Upper Threshold/Pause Packets (Offset: 0x10, width: 32)				
31:13	**Reserved**				
12:0	**Upper Threshold/ Pause Packets**	If this queue is filled up to this Threshold, signal XmtPausePkt of the MAC is asserted (if **En Pause** is set). Multiples of 8 bytes. No effect, if set to ZERO.	yes (for init/ tests only)	aw	0

Bit	Name	Description	Write	Read	Reset (private)
	Receive Rambuffer Lower Threshold/Pause Packets (Offset: 0x14, width: 32)				
31:13	**Reserved**				
12:0	**Lower Threshold/ Pause Packets**	Signal XmtPausePkt of the MAC is deasserted, if the number of bytes falls below Lower Threshold. Multiples of 8 bytes.	yes (for init/ tests only)	aw	0
	Receive Rambuffer Upper Threshold/High Priority (Offset: 0x18, width: 32)				
31:13	**Reserved**				
12:0	**Upper Threshold/ High Priority**	If this queue is filled up to this Threshold, all arbiters grant highest priority to the requests of this queue. Multiples of 8 bytes. No effect, if set to ZERO.	TO	aw	0
	Receive Rambuffer Lower Threshold/High Priority (Offset: 0x1c, width: 32)				
31:13	**Reserved**				
12:0	**Lower Threshold/ High Priority**	All arbiters grant normal priority to the requests of this queue, if the number of bytes falls below Lower Threshold. Multiples of 8 bytes.	yes (for init/ tests only)	aw	0
	Receive Rambuffer Packet Counter (Offset: 0x20, width: 32)				
31:13	**Reserved**				
12:0	**Packet Counter**	Packet Counter gives the current number of packets in this queue.	yes (for tests only)	value	0
	Receive Rambuffer Level (Offset: 0x24, width: 32)				
31:13	**Reserved**				
12:0	**Level**	Level gives the current number of data in multiples of 8 bytes in this queue (including 16 bytes Statusword per packet).	ne	value	0

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Receive Rambuffer Control/Test (Offset: 0x28, width: 32)				
		Test			
31:20	**Reserved**				
19	**Packet Counter Step Down**	Packet Counter decrement	exec	0	
18	**Packet Counter Test On**	Testmode ON/OFF	exec	0b10	0
17	**Packet Counter Test Off**			0b01	1
16	**Packet Counter Step Up**	Packet Counter increment	exec	0	
15	**Reserved**				
14	**Write Pointer Test On**	Testmode ON/OFF	exec	0b10	0
13	**Write Pointer Test Off**	For Testmode ON this rambuffer queue **MUST** be set to NON Operational		0b01	1
12	**Write Pointer Step**	Write Pointer increment	exec	0	
11	**Reserved**				
10	**Read Pointer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Read Pointer Test Off**	For Testmode ON this rambuffer queue **MUST** be set to NON Operational		0b01	1
8	**Read Pointer Step**	Read Pointer decrement	exec	0	
	Control/Status				
7:6	**Reserved**				
5	**St&Fwd On**	Store and Forward On/Off	exec	0b10	0
4	**St&Fwd Off**	If On, a frame is forwarded from Rambuffer only, if the complete frame is in Rambuffer.		0b01	1

Bit	Name	Description	Write	Read	Reset (pri-vate)
3	**Operational On**	Operational mode ON/OFF OFF resets all activities of this rambuffer queue for initialization of the pointers and registers	exec	0b10	0
2	**Operational Off**			0b01	1
1	**Reset Clear**	Set/Clear **Reset.** Executed, if appropriate bit is set to 1. If **Reset** is set, all Receive Rambuffer functions and registers are in their reset state.	exec	0b10	0 (SW)
0	**Reset Set**			0b01	1 (SW)

3.3.2.45 Transmit Rambuffer Registers

Address: 0x0a80 - 0x0aa8 Width [bit]: 7 x 32 (asynchronous transmit queue)

Initialization or re-arrangement of a Rambuffer should ever start from reset state.

Bit	Name	Description	Write	Read	Reset (private)
	Transmit Rambuffer Start Address (Offset: 0x00, width: 32)				
31:13	**Reserved**				
12:0	**Start Address**	Start Address in qwords of this queue in internal memory. Bit [7:0] are replaced by ZEROes internally, which means that the address is limited to multiples of 1 kB. Has to be defined after each Reset.	yes (for init/ tests only)	aw	0
	Transmit Rambuffer End Address (Offset: 0x04, width: 32)				
31:13	**Reserved**				
12:0	**End Address**	End Address in qwords of this queue in internal memory. Bit [7:0] are replaced by ONEs internally, which means that the address is limited to multiples of 1 kB. Has to be defined after each Reset.	yes (for init/ tests only)	aw	0
	Transmit Rambuffer Write Pointer (Offset: 0x08, width: 32)				
31:13	**Reserved**				
12:0	**Write Pointer**	Write Pointer in qwords. Has to be set to **Transmit Rambuffer Start Address** after each Reset.	yes (for init/ tests only)	value	0
	Transmit Rambuffer Read Pointer (Offset: 0x0c, width: 32)				
31:13	**Reserved**				
12:0	**Read Pointer**	Read Pointer in qwords. Has to be set to **Transmit Rambuffer Start Address** after each Reset.	yes (for init/ tests only)	value	0

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Packet Counter (Offset: 0x20, width: 32)				
31:13	**Reserved**				
12:0	**Packet Counter**	Packet Counter provides the current number of packets in this queue.	yes (for tests only)	value	0
	Transmit Rambuffer Level (Offset: 0x24, width: 32)				
31:13	**Reserved**				
12:0	**Level**	Level gives the current number of data in multiples of 8 bytes in this queue (includ-ing 16 bytes Statusword per packet).	ne	value	0
	Rambuffer Control/Test (Offset: 0x28, width: 32)				
		Test			
31:20	**Reserved**				
19	**Packet Counter Step Down**	Packet Counter decrement	exec	0	
18	**Packet Counter Test On**	Testmode ON/OFF	exec	0b10	0
17	**Packet Counter Test Off**			0b01	1
16	**Packet Counter Step Up**	Packet Counter increment	exec	0	
15	**Reserved**				
14	**Write Pointer Test On**	Testmode ON/OFF	exec	0b10	0
13	**Write Pointer Test Off**	For Testmode ON this rambuffer queue **MUST** be set to NON Operational		0b01	1
12	**Write Pointer Step**	Write Pointer increment	exec	0	
11	**Reserved**				

Bit	Name	Description	Write	Read	Reset (pri-vate)
10	**Read Pointer Test On**	Testmode ON/OFF For Testmode ON this rambuffer queue **MUST** be set to NON Operational	exec	0b10	0
9	**Read Pointer Test Off**			0b01	1
8	**Read Pointer Step**	Read Pointer decrement	exec	0	
	Control/Status				
7	**Reserved**				
6	**Reserved**				
5	**St&Fwd On**	Store and Forward On/Off If On, a frame is forwarded from Rambuffer only, if the complete frame is in Rambuffer. If On, this overrides the control on a per frame base as defined by the Transmit Descriptor. Because most PCI systems do not provide Gigabit Ethernet bandwidth, **St&Fwd** should be switched **On** for the Transmit Rambuffer(s).	exec	0b10	0
4	**St&Fwd Off**			0b01	1
3	**Operational On**	Operational mode ON/OFF OFF resets all activities of this Rambuffer queue for initialization of the pointers and registers	exec	0b10	0
2	**Operational Off**			0b01	1
1	**Reset Clear**	Set/Clear **Reset.** Executed, if appropriate bit is set to 1. If **Reset** is set, all Transmit Rambuffer functions and registers are in their reset state.	exec	0b10	0 (SW)
0	**Reset Set**			0b01	1 (SW)

3.3.2.46 Receive MAC FIFOs Registers

Address: 0x0c40 Width [bit]: 7 x 32

Initialization or re-arrangement of a MAC FIFO should ever start from reset state.

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Receive MAC FIFO End Address (Offset: 0x00, width: 32)				
31:7	**Reserved**				
6:0	**End Address**	End Address in qwords of this queue in internal dual port memory. Max = reset value, min = 0x04 Reset value should be used.	yes (for init/ tests only)	aw	0x7f
	Receive MAC FIFO Almost Full Threshold (Offset: 0x04, width: 32)				
31:7	**Reserved**				
6:0	**Almost Full Threshold**	Almost Full Threshold in qwords	yes (for tests only)	value	0x70
	Receive MAC FIFO Control/Test (Offset: 0x08, width: 32)				
31:28	**Reserved**				
27	**Truncation Enable**	Controls truncation of packets within FIFO Default: disabled 0b01	exec	0b10	0
26	**Truncation Dis-able**			0b01	1
25	**VLAN Enable**	Enables/disables VLAN stripping Default: disabled 0b01	exec	0b10	0
24	**VLAN Disable**			0b01	1
23:15	**Reserved**				
14	**Write Pointer Test On**	Testmode ON/OFF	exec	0b10	0
13	**Write Pointer Test Off**			0b01	1
12	**Write Pointer Step**	Write Pointer increment	exec	0	

Bit	Name	Description	Write	Read	Reset (pri-vate)
11	**Reserved**				
10	**Read Pointer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Read Pointer Test Off**			0b01	1
8	**Read Pointer Step**	Read Pointer increment	exec	0	
	Receive MAC FIFO Control				
7	**FIFO Flush On**	FIFO Flush mode ON/OFF	exec	0b10	0
6	**FIFO Flush Off**	When set to ON packets of size below the FIFO Flush Threshold are flushed, if a bit of its Status matches a one in the FIFO Flush Mask. When set to OFF no packets are flushed.		0b01	1
5	**Clear IRQ Receive FIFO Overrun**	Receive FIFO Overrun Interrupt	exec	0	
4	**Clear IRQ Frame Reception Com-plete**	Frame Reception Complete Interrupt	exec	0	
3	**Operational On**	Operational mode ON/OFF	exec	0b10	0
2	**Operational Off**	OFF resets all activities of this FIFO for initialization of the pointers and registers		0b01	1
1	**MAC FIFO Reset Clear**	Set/Clear **MAC FIFO Reset.** Executed, if appropriate bit is set to 1. If **MAC FIFO Reset** is set, all FIFO func-tions and Registers are in their reset state.	exec	0b10	0 (SW)
0	**MAC FIFO Reset Set**			0b01	1 (SW)
	Receive MAC FIFO Flush Mask (Offset: 0x0c, width: 32)				
31:15	**Reserved**				
14	**Length Error**				
13	**VLAN packet**		yes	aw	0
12	**Jabber**		yes	aw	0

Bit	Name	Description	Write	Read	Reset (private)
11	**Undersize packet**		yes	aw	0
10	**Multicast packet**		yes	aw	0
9	**Broadcast packet**		yes	aw	0
8	**Receive OK**		yes	aw	0
7	**Good flow control packet**		yes	aw	0
5	**MII error**		yes	aw	0
4	**Too long packet**		yes	aw	0
3	**Fragment**		yes	aw	0
2	**reserved**		yes	aw	0
1	**CRC error**		yes	aw	0
0	**Receive FIFO overflow**		yes	aw	0
	Receive MAC FIFO Flush Threshold (Offset: 0x10, width: 32)				
31:7	**Reserved**				
6:0	**Flush Threshold**	The Flush threshold is defined in FIFO words (8 bytes).	yes	aw	0x0a
	Receive Truncation Threshold (Offset: 0x14, width: 32)				
31:7	**Reserved**				
6:0	**Truncation Threshold**	The truncation threshold is defined in FIFO words (8 bytes). Loaded with the desired number of receive dwords -2. Example: For the acceptance of only the first ten bytes of a packet a value of eight is written into this register.	yes	aw	0x0a
	Receive VLAN Type Register (Offset: 0x1c, width: 32)				
31:16	**Reserved**				

Bit	Name	Description	Write	Read	Reset (private)
15:0	**Receive VLAN Type**	Code of received VLAN Type. Default: 0x8100 The byte order within the VLAN Type value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).	yes (for tests only)	value	0x8100
	Receive MAC FIFO Write Pointer (Offset: 0x20, width: 32)				
31:7	**Reserved**				
6:0	**Write Pointer**	Write Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Receive MAC FIFO Write Level (Offset: 0x28, width: 32)				
31:7	**Reserved**				
6:0	**Write Level**	Level gives the current number of data in multiples of 8 bytes in this queue (including 16 bytes Statusword per packet).	ne	value	0
	Receive MAC FIFO Read Pointer (Offset: 0x30, width: 32)				
31:7	**Reserved**				
6:0	**Read Pointer**	Read Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Receive MAC FIFO Read Level (Offset: 0x38, width: 32)				
31:7	**Reserved**				
6:0	**Read Level**	Level gives the current number of data in multiples of 8 bytes in this queue (including 16 bytes Statusword per packet).	ne	value	0

3.3.2.47 Transmit MAC FIFOs Registers

Address: 0x0d40 Width [bit]: 10 x 32

Bit	Name	Description	Write	Read	Reset (pri-vate)
	Transmit MAC FIFO End Address (Offset: 0x00, width: 32)				
31:7	**Reserved**				
6:0	**End Address**	End Address in qwords of this queue in internal dual port memory. Max = reset value, min = 0x04 Reset value should be used.	yes (for init/ tests only)	aw	0x7f
	Transmit MAC FIFO Almost Empty Threshold (Offset: 0x04, width: 32)				
31:7	**Reserved**				
6:0	**Almost Empty Threshold**	Almost Empty Threshold in qwords	yes (for tests only)	value	0x10
	Transmit MAC FIFO Control/Test (Offset: 0x08, width: 32)				
31:26	**Reserved**				
25	**VLAN Enable**	Enables/disables VLAN tagging Default: disabled 0b01	exec	0b10	0
24	**VLAN Disable**			0b01	1
23:19	**Reserved**				
18	**Write Shadow Pointer Test On**	Testmode ON/OFF	exec	0b10	0
17	**Write Shadow Pointer Test Off**			0b01	1
16	**Write Shadow Pointer Step**	Write Pointer increment	exec	0	
15	**Reserved**				

Bit	Name	Description	Write	Read	Reset (pri-vate)
14	**Write Pointer Test On**	Testmode ON/OFF	exec	0b10	0
13	**Write Pointer Test Off**			0b01	1
12	**Write Pointer Step**	Write Pointer increment	exec	0	
11	**Reserved**				
10	**Read Pointer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Read Pointer Test Off**			0b01	1
8	**Read Pointer Step**	Read Pointer increment	exec	0	
	Transmit MAC FIFO Control				
7	**Reserved**				
6	**Clear IRQ Transmit FIFO Underrun**	Transmit FIFO Underrun Interrupt	exec	0	
5	**Clear IRQ Frame Transmission Complete**	Frame Transmission Complete Interrupt	exec	0	
4	**Clear IRQ Parity Error**	Clear Parity Error Interrupt	exec	0	
3	**Operational On**	Operational mode ON/OFF	exec	0b10	0
2	**Operational Off**	OFF resets all activities of this FIFO for initialization of the pointers and registers		0b01	1
1	**MAC FIFO Reset Clear**	Set/Clear **MAC FIFO Reset.** Executed, if appropriate bit is set to 1.	exec	0b10	0 (SW)
0	**MAC FIFO Reset Set**	If **MAC FIFO Reset** is set, all FIFO functions and Registers are in their reset state.		0b01	1 (SW)
	Transmit VLAN Type Register (Offset: 0x1c, width: 32)				
31:16	**Reserved**				

Bit	Name	Description	Write	Read	Reset (pri-vate)
15:0	**Transmit VLAN Type**	Code of transmitted VLAN Type. Default: 0x8100 The byte order within the VLAN Type value is big endian: The MSB is in the lower byte (bit 7:0) and the LSB is in the higher byte (bit 15:8).	yes (for tests only)	value	0x8100
	Transmit MAC FIFO Write Pointer (Offset: 0x20, width: 32)				
31:7	**Reserved**				
6:0	**Write Pointer**	Write Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Transmit MAC FIFO Write Shadow Pointer (Offset: 0x24, width: 32)				
31:7	**Reserved**				
6:0	**Write Shadow Pointer**	Write Shadow Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Transmit MAC FIFO Write Level (Offset: 0x28, width: 32)				
31:7	**Reserved**				
6:0	**Write Level**	Level gives the current number of data in multiples of 8 bytes in this queue (including 16 bytes Statusword per packet).	ne	value	0
	Transmit MAC FIFO Read Pointer (Offset: 0x30, width: 32)				
31:7	**Reserved**				
6:0	**Read Pointer**	Read Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Transmit MAC FIFO Restart Pointer (Offset: 0x34, width: 32)				
31:7	**Reserved**				

Bit	Name	Description	Write	Read	Reset (pri-vate)
6:0	**Restart Pointer**	Restart Pointer in qwords of this queue in internal dual port memory.	yes (for tests only)	value	0
	Transmit MAC FIFO Read Level (Offset: 0x38, width: 32)				
31:7	**Reserved**				
6:0	**Read Level**	Level gives the current number of data in multiples of 8 bytes in this queue (including 16 bytes Statusword per packet).	ne	value	0

3.3.2.48 Descriptor Poll Timer Registers

Address: 0x0e00 Width [bit]: 3 x 32

The Poll Timer generates a periodical trigger signal for all BMUs setting **START xxx** such initiating a descriptor read.

This may be enabled for each BMU individually through **En Polling** in the BMU's **Control/Status Registers**.

Bit	Name	Description	Write	Read	Reset (SW)
	Descriptor Poll Timer Init Value				
31:0	**Init Value**	Descriptor Poll Timer Init Value	yes	aw	0
	Descriptor Poll Timer				
31:0	**Descriptor Poll Timer**	Multiples of core clock (125 MHz) Cycle Time T_{Max} = 27.53 s	TO	value	0
	Descriptor Poll Timer Control/Test				
		Test			
31:19	**Reserved**				
18	**Descriptor Poll Timer Test On**	Testmode ON/OFF	exec	0b10	0
17	**Descriptor Poll Timer Test Off**			0b01	1

Bit	Name	Description	Write	Read	Reset (SW)
16	**Descriptor Poll Timer Step**	Timer decrement	exec	0	
15:8	**Reserved**				
		Control			
7:3	**Reserved**				
2	**Descriptor Poll Timer Start**	Start/Stop Timer	exec	0b10	0
1	**Descriptor Poll Timer Stop**			0b01	1
0	**Reserved**				

The timer implements write posting and retries the following accesses to the timer while a posted write is in progress.

Target reads are retried until the addressed register is synchronized to PCI Clock.

3.3.2.49 Timestamp Timer Registers

Address: 0x0e10 Width [bit]: 3 x 32

The Timestamp Timer generates the timebase for the timestamp, which is passed to each frame's descriptor. When wrapping around from 0xffff_ffff to 0x0000_0000 it generates an interrupt.

Bit	Name	Description	Write	Read	Reset (SW)
31:0	**Reserved**				
	Timestamp Timer				
31:0	**Timestamp Timer**	Multiples of core clock (125 MHz) Cycle Time	yes (for tests only)	value	0
	Timestamp Timer Control/Test				
		Test			
31:11	**Reserved**				

Bit	Name	Description	Write	Read	Reset (SW)
10	**Timestamp Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Timestamp Timer Test Off**			0b01	1
8	**Timestamp Timer Step**	Timer increment	exec	0	
		Control			
7:3	**Reserved**				
2	**Timestamp Timer Start**	Start/Stop Timer When started, timer starts counting from value ZERO.	exec	0b10	0
1	**Timestamp Timer Stop**			0b01	1
0	**Timestamp Timer Clear IRQ**	Clear Timer Interrupt	exec	0	

The timer implements write posting and retries the following accesses to the timer while a posted write is in progress.

Target reads are retried until the addressed register is synchronized to PCI Clock.

3.3.2.50 Polling Unit Registers

Address: 0x0e20 Width [bit]: 4 x 32

Bit	Name	Description	Write	Read	Reset (Link)
	Poll Control Register (address: 0x0e20, width: 32)				
31:6	**Reserved**				
5	**Clear IRQ**	Clears the check IRQ generated by the Polling Unit.	exec	0	0
4	**Poll Request**	For testing purpose a Poll Request can be started by SW by writing a 1 to this bit.	exec	0	0

Bit	Name	Description	Write	Read	Reset (Link)
3	**Operational On**	Operational mode ON/OFF OFF resets all activities of the polling unit for initialization of the pointers and registers. The default operation is disabled.	exec	0b10	0
2	**Operational Off**			0b01	1
1	**POLLU Reset Clear**	Set/Clear **Poll Unit Reset.** Executed, if appropriate bit is set to 1. If **Poll Unit Reset** is set, the integrated POLLU and the POLLU Target Interface are in their reset state.	exec	0b10	0
0	**POLLU Reset Set**			0b01	1
List Last Index (address: 0x0e24, width: 12))					
31:12	**Reserved**				
11:0	**List Last Index**	Specifies the index of the last entry in the list. The descriptor element length is defined by List Last Index + 1. The length is given as number of list elements. The length is fixed. Only 2 LE (dual link version) are read each time.	no	0x001	0x001
List Start Address Low (address: 0x0e28, width: 32)					
31:3		Identifies the lower 32 bit of the start address of the descriptor list in the system memory.	yes	aw	0x0
2:0		The address is always aligned to Descriptor List Element size, i.e. 8 byte boundaries, i.e. the lower bits are always 0x0.	ne	0x0	0x0
List Start Address High (address: 0x0e2c, width: 32)					
31:0		Identifies the upper 32 bit of the start address of the descriptor list in the system memory.	yes	aw	0x0

3.3.2.51 Status Unit Registers

Address: 0x0e80 Width [bit]: 24 x 32

Bit	Name	Description	Write	Read	Reset (SW)
Status Unit Registers					
	Status BMU Control (offset: 0x00, width: 32)				
31:19	**Reserved**				
18	**Master Request Loopback Test On**	Testmode ON/OFF for the Master request, i.e. enables the stepping of the Master request by the SW. The default operation is disabled.	exec	0b10	0
17	**Master Request Loopback Test Off**			0b01	1
16	**Master Request Loopback Step**	Enables the Loopback of a single master request. For execution of this command, the Master Request Loopback Test **MUST** be set to ON.	exec	0	0
15	**Reserved**				
14	**FIFO Read Pointer Test On**	Testmode ON/OFF for the FIFO Read Pointer, i.e. enables the stepping of the pointer by the SW. The default operation is disabled.	exec	0b10	0
13	**FIFO Read Pointer Test Off**			0b01	1
12	**FIFO Read Pointer Step**	FIFO Read Pointer increment. For execution of this command, the FIFO Read Pointer Test **MUST** be set to ON.	exec	0	0
11	**Reserved**				
10	**FIFO Write Pointer Test On**	Testmode ON/OFF for the FIFO Write Pointer, i.e. enables the stepping of the pointer by the SW. The default operation is disabled.	exec	0b10	0
9	**FIFO Write Pointer Test Off**			0b01	1
8	**FIFO Write Pointer Step**	FIFO Write Pointer increment. For execution of this command, the FIFO Write Pointer Test **MUST** be set to ON.	exec	0	0
7:5	**Reserved**				

Bit	Name	Description	Write	Read	Reset (SW)
4	**Clear IRQ Status BMU**	Clear Status BMU interrupt	exec	0	
3	**Operational On**	Default Off	exec	0b10	0
2	**Operational Off**			0b01	1
1	**Enable**	Default Reset	exec	0b10	0
0	**Reset**			0b01	1
	Last Index (offset: 0x04, width: 16)				
31:12	**Reserved**				
11:0	**Last Index**	Index of last element of current list Bit 6:0: fixed value 0x7F List length is a multiple of 128 elements.	yes	aw	0x007F
	List Start Address (offset: 0x08, width: 2 x 32)				
31:0	**List Start Address Lo**	Lower part of list start address Bit 11:0: fixed value 0x000 Address aligned to 4 KByte boundaries.	yes	aw	0
31:0	**List Start Address Hi**	Higher part of list start address	yes	aw	0
	TXA Report Index (offset: 0x10, width: 16)				
15:12	**Reserved**				
11:0	**TXA Report Index**	Last reported TX Done Index for TXA queue	ne	value	0
15:0	**Reserved**				
	TX Index Threshold (offset: 0x18, width: 16)				
15:12	**Reserved**				
11:0	**TX Index Threshold**	A status burst is initiated when exceeding the threshold.	yes	aw	0xFFF

Bit	Name	Description	Write	Read	Reset (SW)
Put Index (Offset: 0x1c, Width: 16)					
31:12	**Reserved**				
11:0		Specifies the Push/Wr address within the descriptor list in system memory. This Put Index points to the address of the next free element in the descriptor list (i.e. the position after the last valid element in the descriptor list). If Get Index and Put Index are equal, no element is in the FIFO. The index is given in list element granularity.	yes	aw	0x0
	FIFO Write Pointer (offset: 0x20, width: 8)				
31:6	**Reserved**				
5:0		FIFO Write Pointer (64 qword FIFO)	TO	value	0
	FIFO Read Pointer (offset: 0x24, width: 8)				
31:6	**Reserved**				
5:0		FIFO Read Pointer (qword)	TO	value	0
	FIFO Level (offset: 0x28, width: 8)				
31:6	**Reserved**				
5:0		FIFO Level (qword)	ne	value	0
	FIFO Watermark (offset: 0x2c, width: 8)				
7:6	**Reserved**				
5:0		FIFO Watermark (qword)	yes	aw	0x30
	FIFO ISR Watermark (offset: 0x2d, width: 8)				
7:6	**Reserved**				
5:0		FIFO ISR Watermark (qword) Watermark relevant during interrupt service routine.	yes	aw	0x08
	Level Timer				
	Level Timer Init Value (offset: 0x30, width: 32)				
31:0	**Level Timer Init Value**	Level Timer Init Value	yes	aw	0x3d09

Bit	Name	Description	Write	Read	Reset (SW)
	Level Timer Counter (offset: 0x34, width: 32)				
31:0	**Level Timer**	Multiples of core clock (125 MHz) Cycle Time T_{Max} = 27.53 s	TO	value	0
	Level Timer Control/Test (offset: 0x38, width: 32)				
		Test			
31:11	**Reserved**				
10	**Level Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**Level Timer Test Off**			0b01	1
8	**Level Timer Step**	Timer decrement	exec	0	
		Control			
7:3	**Reserved**				
2	**Level Timer Start**	Start/Stop Level Timer	exec	0b10	0
1	**Level Timer Stop**			0b01	1
0	**Reserved**				
	TX Timer				
	TX Timer Init Value (offset: 0x40, width: 32)				
31:0	**TX Timer Init Value**	TX Timer Init Value	yes	aw	0
	TX Timer Counter (offset: 0x44, width: 32)				
31:0	**TX Timer**	Multiples of core clock (125 MHz) Cycle Time T_{Max} = 27.53 s	TO	value	0
	TX Timer Control/Test (offset: 0x48, width: 32)				
		Test			
31:11	**Reserved**				
10	**TX Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**TX Timer Test Off**			0b01	1
8	**TX Timer Step**	Timer decrement	exec	0	

Bit	Name	Description	Write	Read	Reset (SW)
		Control			
7:3	**Reserved**				
2	**TX Timer Start**	Start/Stop TX Timer	exec	0b10	0
1	**TX Timer Stop**			0b01	1
0	**Reserved**				
	ISR Timer				
	ISR Timer Init Value (offset: 0x50, width: 32)				
31:0	**ISR Timer Init Value**	ISR Timer Init Value	yes	aw	0
	ISR Timer Counter (offset: 0x54, width: 32)				
31:0	**ISR Timer**	Multiples of core clock (125 MHz) Cycle Time T_{Max} = 27.53 s	TO	value	0
	ISR Timer Control/Test (offset: 0x58, width: 32)				
		Test			
31:11	**Reserved**				
10	**ISR Timer Test On**	Testmode ON/OFF	exec	0b10	0
9	**ISR Timer Test Off**			0b01	1
8	**ISR Timer Step**	Timer decrement	exec	0	
		Control			
7:3	**Reserved**				
2	**ISR Timer Start**	Start/Stop ISR Timer	exec	0b10	0
1	**ISR Timer Stop**			0b01	1
0	**Reserved**				

3.3.2.52 MAC Control Registers

Address: 0x0f00 Width [bit]:8

Bit	Name	Description	Write	Read	Reset (Link)
	MAC Control Register				
15:8	**Reserved**				
7	**En Burstmode On**	Enables the FIFO to signal GMAC, that enough packets are available to enter Half Duplex Burst mode.	exec	0b10	0
6	**En Burstmode Off**			0b01	1
5	**Loopback On**	The GMAC Interface may be set to loop-back mode for testing purposes. Transmit data is looped to receive data. There is no activity at the related GMAC control signals. The Transmit Status (descriptors) MUST be set to the expected Receive Status (descriptors).	exec	0b10	0
4	**Loopback Off**			0b01	1
3	**En Pause On**	Enable forwarding of the signal Xmt-PausePkt to GMAC see also **Receive Rambuffer Upper/ Lower Threshold/Pause Packets** Must be switched OFF in loopback mode. Settings for GMAC MUST be consistent.	exec	0b10	0
2	**En Pause Off**			0b01	1
1	**GMAC Reset Clear**	Set/Clear **GMAC Reset.** Executed, if appropriate bit is set to 1. If **GMAC Reset** is set, the integrated GMAC and the GMAC Target Interface are in their reset state. In order to guarantee a minimum pulse width of 31 core clock cycles, clearing **GMAC Reset** is suppressed within 31 core clock cycles after **GMAC Reset Set**. Write cycles to **GMAC Reset Clear** with-in the 31 core clock recovery time, are terminated with a Target Retry (no impact on software).	exec	0b10	0
0	**GMAC Reset Set**			0b01	1

3.3.2.53 PHY MUX Register and PHY Control Register

Address: 0x0f04 Width [bit]:8 + 24

The PHY Control Register contains settings for the PHY.

The meaning of the seven configuration pins is described in "88E1111 Integrated 10/100/1000 Gigabit Ethernet Transceiver", Marvell, Doc. No. MV-S100707-00 - Hardware Configuration.

Bit	Name	Description	Write	Read	Reset (Link)
	PHY MUX Register, loaded from external memory				
31:29	**Reserved**				
28:26	**MUX6**	3 Bit Multiplexer Setting for Config6 Pin	yes	aw	0b000
25:23	**MUX5**	3 Bit Multiplexer Setting for Config5 Pin	yes	aw	0b000
22:20	**MUX4**	3 Bit Multiplexer Setting for Config4 Pin	yes	aw	0b000
19:17	**MUX3**	3 Bit Multiplexer Setting for Config3 Pin	yes	aw	0b000
16:14	**MUX2**	3 Bit Multiplexer Setting for Config2 Pin	yes	aw	0b000
13:11	**MUX1**	3 Bit Multiplexer Setting for Config1 Pin	yes	aw	0b000
10:8	**MUX0**	3 Bit Multiplexer Setting for Config0 Pin	yes	aw	0b000
	PHY Control Register				
7:2	**Reserved**				
1	**PHY Reset Clear**	Set/Clear **PHY Reset.** Executed, if appropriate bit is set to 1. If **PHY Reset** is set, the integrated PHY is in its reset state. In order to guarantee a minimum pulse width of 31 core clock cycles, clearing **PHY Reset** is suppressed within 31 core clock cycles after **PHY Reset Set**. Write cycles to **PHY Reset Clear** within the 31 core clock recovery time, are terminated with a Target Retry (no impact on software).	exec	0b10	0
0	**PHY Reset Set**			0b01	1

Table 28: Pin to Constant Mapping (MUX to CONFIG pins)

PIN	BIT<2:0>
VDD (=1)	111
LED_LINK10/100	101
LED_LINK1000	100
LED_LINKn	011
LED_ACTn	010
VSS (=0)	000

3.3.2.54 MAC Interrupt Source Register

Address:0x0f08 Width [bit]:8
If set to one, interrupt is pending.

Bit	Name	Description	Write	Read	Reset (SW)
MAC Interrupt Source Register					
31:6	**Reserved**				
5	**Transmit Counter Overflow Interrupt**	One or more bits are set in the **Transmit Counter Interrupt Register**	ne	value	0
4	**Receive Counter Overflow Interrupt**	One or more bits are set in the **Receive Counter Interrupt Register**.	ne	value	0
3	**Transmit FIFO Underrun**	Underrun condition in the Tx MAC FIFO	ne	value	0
2	**Frame Transmission Complete**	Frame Transmission Complete. Frame is copied from the Tx MAC FIFO and transmitted by the MAC successfully	ne	value	0
1	**Receive FIFO Overrun**	Overflow condition in the Rx MAC FIFO	ne	value	0
0	**Frame Reception Complete**	Frame Reception Complete. Frame is copied into the Rx MAC FIFO successfully	ne	value	0

3.3.2.55 MAC Interrupt Mask Register

Address: 0x0f0c Width [bit]:8
The enable bits have the same bit positions as in the MAC Interrupt Source Register.
If set to one, interrupt is enabled.

Bit	Name	Description	Write	Read	Reset (SW)
	MAC Interrupt Mask Register				
31:0	**En IRQ xxx**	Enable Interrupt xxx	yes	aw	0

3.3.2.56 Link Control Register

Address: 0x0f10 Width [bit]:8
The Link Control Register contains settings for the parts needed to establish a link.

Bit	Name	Description	Write	Read	Reset (Power on)
	Link Control Register				
15:2	**Reserved**				
1	**Link Reset Clear**	Set/Clear **Link Reset.** Executed, if appropriate bit is set to 1. If **Link Reset** is set, the **WOL Unit**, the **MAC Control Register** and the **PHY Control Register** are in their reset state. This also implies that the integrated GMAC and PHY are also in their reset state.	exec	0b10	0
0	**Link Reset Set**			0b01	1

3.3.2.57 Wake on LAN Control Registers

Base address: 0x0f20 Width [bit]:8 x 32
The Wake on LAN Control Registers are used to control the Wake up Frame Unit, the Magic Pattern Unit and the Link Change Unit.

Bit	Name	Description	Write	Read	Reset (Link)
	Match Result Register (offset: 0x03, width: 8)				
31	**Reserved**				
30	**Pattern 6 Match**	If set, the last incoming packet matched with pattern number <x>	ne	value	0b0
29	**Pattern 5 Match**		ne	value	0b0
28	**Pattern 4 Match**		ne	value	0b0
27	**Pattern 3 Match**		ne	value	0b0
26	**Pattern 2 Match**		ne	value	0b0
25	**Pattern 1 Match**		ne	value	0b0
24	**Pattern 0 Match**		ne	value	0b0
	Match Control Register (offset: 0x02, width: 8)				
23	**Reserved**				
22	**Pattern 6 Enable**	If set, pattern number <x> is compared with incoming packets	yes	aw	0b0
21	**Pattern 5 Enable**		yes	aw	0b0
20	**Pattern 4 Enable**		yes	aw	0b0
19	**Pattern 3 Enable**		yes	aw	0b0
18	**Pattern 2 Enable**		yes	aw	0b0
17	**Pattern 1 Enable**		yes	aw	0b0
16	**Pattern 0 Enable**		yes	aw	0b0
	WOL Control/Status Register (offset: 0x00, width: 16)				
15	**Link Change Status**	Set, if a Link Change wake up event occurred	ne	value	0
14	**Magic Pattern Status**	Set, if a Magic Pattern wake up event occurred	ne	value	0
13	**Wake up Frame Status**	Set, if a Wake up Frame wake up event occurred	ne	value	0

Bit	Name	Description	Write	Read	Reset (Link)
12	**Clear Match Result**	Clears the **Magic Pattern Status** and **Link Change Status** bits in the **WOL Control/Status Register** and the Match bits in the **Match Result Register**.	exec	0	
11	**Enable PME on Link Change**	Enable/Disable **PME on Link Change.** Executed, if appropriate bit is set to 1. If **PME on Link Change** is enabled and **PME EN** in the PCI Power Management Control Register is set, **PME#** on the PCI bus is asserted on a successful pattern match in the Link Change unit.	exec	0b10	0
10	**Disable PME on Link Change**			0b01	1
9	**Enable PME on Magic Pattern**	Enable/Disable **PME on Magic Pattern.** Executed, if appropriate bit is set to 1. If **PME on Magic Pattern** is enabled and **PME EN** in the PCI Power Management Control Register is set, **PME#** on the PCI bus is asserted on a successful pattern match in the Magic Pattern unit.	exec	0b10	0
8	**Disable PME on Magic Pattern**			0b01	1
7	**Enable PME on Wake up Frame**	Enable/Disable **PME on Wake up Frame.** Executed, if appropriate bit is set to 1. If **PME on Wake up Frame** is enabled and **PME EN** in the PCI Power Management Control Register is set, **PME#** on the PCI bus is asserted on a successful pattern match in the Wake up Frame unit.	exec	0b10	0
6	**Disable PME on Wake up Frame**			0b01	1
5	**Enable Link Change Unit**	Enable/Disable **Link Change Unit.** Executed, if appropriate bit is set to 1. If **Link Change Unit** is enabled, on each Link Up Event the **Link Change Status** bit in the **WOL Control/Status Register** is set and depending on the PME settings **PME#** is asserted.	exec	0b10	0
4	**Disable Link Change Unit**			0b01	1
3	**Enable Magic Pattern Unit**	Enable/Disable **Magic Pattern Unit.** Executed, if appropriate bit is set to 1. If **Magic Pattern Unit** is enabled, each incoming packet is compared to the **AMD Magic Pattern**. If a pattern match occurs the corresponding **Magic Pattern Status** bit in the **WOL Control/Status Register** is set and depending on the PME settings **PME#** is asserted.	exec	0b10	0
2	**Disable Magic Pattern Unit**			0b01	1

Bit	Name	Description	Write	Read	Reset (Link)
1	**Enable Wake up Frame Unit**	Enable/Disable **Wake up Frame Unit.** Executed, if appropriate bit is set to 1. If **Wake up Frame Unit** is enabled, each incoming packet is compared to each of the 7 wake up pattern stored in the Pattern RAM that has its corresponding **Pattern Enable** bit set. If a pattern match occurs the corresponding **Pattern Match** bit in the **Match Result Register** is set and depending on the PME settings **PME#** is asserted.	exec	0b10	0
0	**Disable Wake up Frame Unit**			0b01	1
	Reserved				
31	**Reserved**				
30	**Reserved**		yes	aw	0
29	**Reserved**		yes	aw	0
28	**Reserved**		yes	aw	0
27	**Reserved**		yes	aw	0
26	**Reserved**		yes	aw	0
25	**Reserved**		yes	aw	0
24	**Reserved**		yes	aw	0
	PME Match Enable Register (offset: 0x0a, width: 8)				
23	**Force PME**	Generates a PME event	exec	0	0
22	**PME 6 Enable**	If set, incoming packets that match with pattern number <x> will generate a PME event.	yes	aw	0
21	**PME 5 Enable**		yes	aw	0
20	**PME 4 Enable**		yes	aw	0
19	**PME 3 Enable**		yes	aw	0
18	**PME 2 Enable**		yes	aw	0
17	**PME 1 Enable**		yes	aw	0
16	**PME 0 Enable**		yes	aw	0
	MAC Address Register High (offset: 0x08, width: 16)				
15:8	**MAC<5>**	Mac-Address, Byte 5	yes	aw	0
7:0	**MAC<4>**	Mac-Address, Byte 4	yes	aw	0

Bit	Name	Description	Write	Read	Reset (Link)
	MAC-Address Registers low (offset: 0x04, width: 32)				
31:24	**MAC<3>**	Mac-Address, Byte 3	yes	aw	0
23:16	**MAC<2>**	Mac-Address, Byte 2	yes	aw	0
15:8	**MAC<1>**	Mac-Address, Byte 1	yes	aw	0
7:0	**MAC<0>**	Mac-Address, Byte 0	yes	aw	0
	Pattern Read Pointer Register (offset: 0x0c, width: 7)				
31:7	**Reserved**				
6:0	**Pattern Read Pointer**	Address of the current 128 bit word in Pattern RAM to be compared, can be written for test purposes only	yes, for Test only	value	0x00
	Pattern Length Register 0 (0 - 3) (offset: 0x10, width: 4 x 8)				
31	**Reserved**				
30:24	**Pattern 3 Length**	Number of bytes - 1 to compare for Pattern 3 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes	yes	aw	0x00
23	**Reserved**				
22:16	**Pattern 2 Length**	Number of bytes - 1 to compare for Pattern 2 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes.	yes	aw	0x00
15	**Reserved**				
14:8	**Pattern 1 Length**	Number of bytes - 1 to compare for Pattern 1 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes.	yes	aw	0x00
7	**Reserved**				

Bit	Name	Description	Write	Read	Reset (Link)
6:0	**Pattern 0 Length**	Number of bytes - 1 to compare for Pattern 0 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes	yes	aw	0x00
	Pattern Length Register 1 (4 - 6) (offset: 0x14, width: 3 x 8)				
31:23	**Reserved**				
22:16	**Pattern 6 Length**	Number of bytes - 1 to compare for Pattern 6 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes	yes	aw	0x00
15	**Reserved**				
14:8	**Pattern 5 Length**	Number of bytes - 1 to compare for Pattern 5. When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes	yes	aw	0x00
7	**Reserved**				
6:0	**Pattern 4 Length**	Number of bytes - 1 to compare for Pattern 4 When pattern length is 64 bytes then value 63 is written to this register. Minimum allowed pattern length is 6 bytes	yes	aw	0x00
	Pattern Counter Register 0 (0 - 3) (offset: 0x18, width: 4 x 8)				
31	**Reserved**				
30:24	**Pattern 3 Counter**	Current byte to compare for Pattern 3	TO	value	0x00
23	**Reserved**				
22:16	**Pattern 2 Counter**	Current byte to compare for Pattern 2	TO	value	0x00
15	**Reserved**				
14:8	**Pattern 1 Counter**	Current byte to compare for Pattern 1	TO	value	0x00
7	**Reserved**				

Bit	Name	Description	Write	Read	Reset (Link)
6:0	**Pattern 0 Counter**	Current byte to compare for Pattern 0	TO	value	0x00
	Pattern Counter Register 1 (4 - 6) (offset: 0x1c, width: 3 x 8)				
31:23	**Reserved**				
22:16	**Pattern 6 Counter**	Current byte to compare for Pattern 6	TO	value	0x00
15	**Reserved**				
14:8	**Pattern 5 Counter**	Current byte to compare for Pattern 5	TO	value	0x00
7	**Reserved**				
6:0	**Pattern 4 Counter**	Current byte to compare for Pattern 4	TO	value	0x00

3.3.2.58 Pattern RAM

Address: 0x1000 - 0x13fc (link 1) Width [bit]:256 x 32 bit
For accesses to the Pattern RAM the WOL pattern matching must be disabled before (set bits 1:0 of WOL Control Register to 0b01 (inactive)).

The Pattern RAM is fully mapped into the I/O space and occupies 256 consecutive 32 bit words. During read accesses data is read directly from RAM and no special sequence is needed. For write accesses the four 32 bit dwords of a 128 bit RAM word are collected in a 128 bit register and the write access to the uppermost dword (address: 0x0c, 0x1c,...) causes the register contents to be written into the RAM. Therefore the lower three dwords have to be written first. The Pattern RAM contents are undetermined after power on.

Bit	Name	Description	Write	Read	Reset (None)
	Pattern RAM				
31:0	**Lower Half qword No. 0**	(offset: 0x00, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx
31:0	**Upper Half qword No. 0**	(offset: 0x04, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx
31:0	**Lower Half qword No. 1**	(offset: 0x08, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx

Bit	Name	Description	Write	Read	Reset (None)
31:0	**Upper Half qword No. 1**	(offset: 0x0c, width: 32 bit) Read reads RAM directly, Write triggers write of 128 bit in RAM	yes	aw	0bxxxx xxxx
	...				
31:0	**Lower Half qword No. 126**	(offset: 0x3f0, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx
31:0	**Upper Half qword No. 126**	(offset: 0x3f4, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx
31:0	**Lower Half qword No. 127**	(offset: 0x3f8, width: 32 bit) Read reads RAM directly, Write writes only intermediate Register	yes	aw	0bxxxx xxxx
31:0	**Upper Half qword No. 127**	(offset: 0x3fc, width: 32 bit) Read reads RAM directly, Write triggers write of 128 bit in RAM	yes	aw	0bxxxx xxxx

3.3.2.59 TCP Segmentation Header Registers

There are four sets of 34 TCP Segmentation Header Registers:

TCP Segmentation Header Registers:

Address: 0x1900 - 0x1984 (asynchronous) Width [bit]: 34 x 32

All sets of Header Registers have the same structure.

All registers are written and read by the TCP Segmentation logic automatically. They can be accessed (read/write) through YTB for debug purposes.

Bit	Name	Description	Write	Read	Reset (Link)
TCP Segmentation Header Register 0 Lo (offset: 0x00 width: 32)					
31:0	**MAC_HDR_0**	Word 0/1 of MAC Header	yes	value	0x0
TCP Segmentation Header Register 0 Hi (offset: 0x04 width: 32)					
31:0	**MAC_HDR_1**	Word 2/3 of MAC Header	yes	value	0x0
TCP Segmentation Header Register 1 Lo (offset: 0x08 width: 32)					
31:0	**MAC_HDR_2**	Word 4/5 of MAC Header	yes	value	0x0

Bit	Name	Description	Write	Read	Reset (Link)
TCP Segmentation Header Register 1 Hi (offset: 0x0c width: 32)					
31:16	**IP_HDR_0**	Word 0 of IP Header	yes	value	0x0
15:0	**MAC_HDR_3**	Word 6 of MAC Header	yes	value	0x0
TCP Segmentation Header Register 2 Lo (offset: 0x10 width: 32)					
31:0	**IP_HDR_1**	Word 1/2 of IP Header	yes	value	0x0
TCP Segmentation Header Register 2 Hi (offset: 0x14 width: 32)					
31:0	**IP_HDR_2**	Word 3/4 of IP Header	yes	value	0x0
TCP Segmentation Header Register 3 Lo (offset: 0x18 width: 32)					
31:0	**IP_HDR_3**	Word 5/6 of IP Header	yes	value	0x0
TCP Segmentation Header Register 3 Hi (offset: 0x1c width: 32)					
31:0	**IP_HDR_4**	Word 7/8 of IP Header	yes	value	0x0
TCP Segmentation Header Register 4 Lo (offset: 0x20 width: 32)					
31:16	**IP_OPT_0**	Word 0 of IP Header Options[1]	yes	value	0x0
15:0	**IP_HDR_5**	Word 9 of IP Header	yes	value	0x0
TCP Segmentation Header Register 4 Hi (offset: 0x24 width: 32)					
31:0	**IP_OPT_1**	Word 1/2 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 5 Lo (offset: 0x28 width: 32)					
31:0	**IP_OPT_2**	Word 3/4 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 5 Hi (offset: 0x2c width: 32)					
31:0	**IP_OPT_3**	Word 5/6 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 6 Lo (offset: 0x30 width: 32)					
31:0	**IP_OPT_4**	Word 7/8 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 6 Hi (offset: 0x34 width: 32)					
31:0	**IP_OPT_5**	Word 9/10 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 7 Lo (offset: 0x38 width: 32)					
31:0	**IP_OPT_6**	Word 11/12 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 7 Hi (offset: 0x3c width: 32)					
31:0	**IP_OPT_7**	Word 13/14 of IP Header Options	yes	value	0x0

Bit	Name	Description	Write	Read	Reset (Link)
TCP Segmentation Header Register 8 Lo (offset: 0x40 width: 32)					
31:0	**IP_OPT_8**	Word 15/16 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 8 Hi (offset: 0x44 width: 32)					
31:0	**IP_OPT_9**	Word 17/18 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 9 Lo (offset: 0x48 width: 32)					
31:16	**TCP_HDR_0**	Word 0 of TCP Header	yes	value	0x0
15:0	**IP_OPT_10**	Word 19 of IP Header Options	yes	value	0x0
TCP Segmentation Header Register 9 Hi (offset: 0x4c width: 32)					
31:0	**TCP_HDR_1**	Word 1/2 of TCP Header	yes	value	0x0
TCP Segmentation Header Register 10 Lo (offset: 0x50 width: 32)					
31:0	**TCP_HDR_2**	Word 3/4 of TCP Header	yes	value	0x0
TCP Segmentation Header Register 10 Hi (offset: 0x54 width: 32)					
31:0	**TCP_HDR_3**	Word 5/6 of TCP Header	yes	value	0x0
TCP Segmentation Header Register 11 Lo (offset: 0x58 width: 32)					
31:0	**TCP_HDR_4**	Word 7/8 of TCP Header	yes	value	0x0
TCP Segmentation Header Register 11 Hi (offset: 0x5c width: 32)					
31:16	**TCP_OPT_0**	Word 0 of TCP Header Options[2]	yes	value	0x0
15:0	**TCP_HDR_5**	Word 9 of TCP Header	yes	value	0x0
TCP Segmentation Header Register 12 Lo (offset: 0x60 width: 32)					
31:0	**TCP_OPT_1**	Word 1/2 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 12 Hi (offset: 0x64 width: 32)					
31:0	**TCP_OPT_2**	Word 3/4 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 13 Lo (offset: 0x68 width: 32)					
31:0	**TCP_OPT_3**	Word 5/6 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 13 Hi (offset: 0x6c width: 32)					
31:0	**TCP_OPT_4**	Word 7/8 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 14 Lo (offset: 0x70 width: 32)					
31:0	**TCP_OPT_5**	Word 9/10 of TCP Header Options	yes	value	0x0

Bit	Name	Description	Write	Read	Reset (Link)
TCP Segmentation Header Register 14 Hi (offset: 0x74 width: 32)					
31:0	**TCP_OPT_6**	Word 11/12 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 15 Lo (offset: 0x78 width: 32)					
31:0	**TCP_OPT_7**	Word 13/14 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 15 Hi (offset: 0x7c width: 32)					
31:0	**TCP_OPT_8**	Word 15/16 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 16 Lo (offset: 0x80 width: 32)					
31:0	**TCP_OPT_9**	Word 17/18 of TCP Header Options	yes	value	0x0
TCP Segmentation Header Register 16 Hi (offset: 0x84 width: 32)					
31:16	**Reserved**		0x0	0x0	0x0
15:0	**TCP_OPT_10**	Word 19 of TCP Header Options	yes	value	0x0

1. The IP Header Option registers are optional. If the IP Header Options don't exist at all or only partially exist, the TCP Header registers and TCP Header Option registers move up accordingly.
2. The TCP Header Option registers are optional. If the TCP Header Options don't exist at all or only partially exist, the unused TCP Header Option registers are undefined.

3.3.2.60 PCI Configuration Register File

Address: 0x1c00 - 0x1efc

The whole Configuration Register File is made accessible within this address area of the control register file.

It is read only with some exceptions: **Our Register 1** and **2** and **VPD Address** and **VPD Data Register** may be written.

Write operations are completed normally on the bus and the data is discarded.

For testing purposes, the Configuration Register File may be set writable by **En Config Write**.

3.4 GMAC Registers

All GMAC Registers are mapped into a subsequent range on 32-bit word boundaries. The GMAC interface implements write posting and delayed transactions on read accesses. Target reads are retried until the addressed register is synchronized to PCI Clock. The GMAC's Node Processor Interface is running in 16-bit mode. The offset of the register addresses in the PCI address space is calculated by multiplying the 16-bit XMAC register addresses by 2.

The GMAC Registers can be accessed with any width at dword boundaries. On the PCI side all accesses to the GMAC are completed normally. **MAC Receive Status Word**

Receive Status Word:

Bit(s)	Description
31:16	Byte Count, this field provides byte count of received packet
15:14	Reserved
13	VLAN packet
12	Jabber (a packet that is too long and has a CRC error)
11	Undersize packet (a packet which is less than 64 bytes, but with a good CRC)
10	Multicast packet
9	Broadcast packet
8	Receive OK (good packet)
7	Good flow control packet
6	Bad flow control packet
5	MII error
4	Oversize (packet longer than max. length with good CRC)
3	Fragment
2	Reserved
1	CRC error
0	Receive FIFO overflow

3.4.1 MAC Register Definitions

The registers in the 88E8053 device are accessible through the CPU interface. The SMI (Serial Management Interface) Control Register, SMI data register, and the PHY Address Register are used to read from and write to registers in the PHY.

Table 29: General Purpose Status Register (GPSR)
Offset: 0x2800

(All valid bits in this register are read-only)

Bits	Field		Type/ Reset Value	Description
15	Speed	**Read**		Port Speed This bit is read-only. 0 - 10 Mbps 1 - 100 Mbps (only valid if GigSpeed bit is 0)
		Write		
		Reset	1	
14	Duplex	**Read**		Port Duplex Mode This bit is read-only. 0 - Half duplex 1 - Full duplex
		Write		
		Reset	0	
13	FctlTx	**Read**		Transmit Flow Control Mode This bit is read-only. 0 - Flow Control Mode enabled 1 - Flow Control Mode disabled
		Write		
		Reset	0	
12	Link	**Read**		Link Status This bit is read-only. 0 - Link is down 1 - Link is up
		Write		
		Reset	1	
11	Pause	**Read**		Port is in "Flow Control Disabled" state, i.e. the transmit state machine is in Pause state. This bit is set when an IEEE 802.3x flow-control PAUSE (XOFF) packet is received (and flow-control is enabled and the port is in full-duplex mode). Reset when XON is received, or when the XOFF timer has expired. This bit is read-only.
		Write		
		Reset	0	
10	TxinProg	**Read**		TX In Progress Indicates that the port's transmitter is in an active transmission state. This bit is read-only.
		Write		
		Reset	0	
9	ExcessCol	**Read**		Excessive Collisions Occurred Indicates that a packet transmission experienced 16 collisions.
		Write		
		Reset	0	
8	LateCol	**Read**		Late Collision Occurred A collision occurred beyond 64-bit-times from start of the packet.
		Write		
		Reset	0	
7:6	Reserved	**Read**		
		Write		
		Reset	0	

Table 29: General Purpose Status Register (GPSR)
Offset: 0x2800

(All valid bits in this register are read-only)

Bits	Field		Type/ Reset Value	Description
5	MIIPhySTC	**Read**		MII PHY Status Change Indicates a status change reported by the PHY connected to this port. Set, when the MII management interface block identifies a change in PHY's register 1.
		Write		
		Reset	0	
4	GigSpeed	**Read**		This bit is only valid if bit 15 (Speed) is set. 0 - follow the Speed bit setting 1 - 1000Mbps operation mode
		Write		
		Reset	1	
3	Partition	**Read**		Partition mode Indicates that the MAC has entered the Partition Mode. When in this mode, the port transmits pending packets, ignoring the collisions.
		Write		
		Reset	0	
2	FctlRx	**Read**		Receive Flow Control Mode This bit is read-only. 0 - Flow-control mode enabled. 1 - Flow-control mode disabled.
		Write		
		Reset	0	
1	Promiscuous Mode	**Read**		This bit is set if the device is in Promiscuous mode. This bit will be set to 1 at power up.
		Write		
		Reset	0	
0	Reserved	**Read**		Reserved
		Write		
		Reset	0	

Table 30: General Purpose Control Register (GPCR)
Offset: 0x2804

Bits	Field		Type/ Reset Value	Description
15:14	Reserved	**Read**		
		Write		
		Reset		
13	FCTLTX	**Read**		Transmit Flow Control Mode 0 - Enable IEEE 802.3x flow control 1 - Disable IEEE 802.3x flow control NOTE: Valid only if auto-negotiation for flow control is disabled.
		Write		
		Reset		

Table 30: General Purpose Control Register (GPCR)
Offset: 0x2804

Bits	Field		Type/ Reset Value	Description
12	TxEn	**Read**		Transmit Enable 0 - Disabled 1 - Enable Ethernet port is ready to transmit.
		Write		
		Reset	0	
11	RxEn	**Read**		Receive Enable 0 - Disabled 1 - Enable Ethernet port is ready to receive.
		Write		
		Reset	0	
10	Reserved	**Read**		
		Write		
		Reset		
9	LPBK	**Read**		Loopback mode 0 - Normal mode 1 - Internal loopback mode Tx data is looped back to the Rx lines and also transmitted to the MII interface pins.
		Write		
		Reset	0	
8	PAR	**Read**		Partition Enable When more than 61 collisions occur while transmitting, the port enters Partition Mode. It waits for the first good packet from the wire, and then goes back to Normal Mode. In Partition Mode it continues transmitting, but it does not receive. 0 - Normal Mode 1 - Partition Mode
		Write		
		Reset	0	
7	GigSpeed	**Read**		GigSpeed 0 - follow the Speed bit setting 1 - 1000Mbps (only valid if Speed bit (3) is set to 1).
		Write		
		Reset	1	
6	FLP	**Read**		Force Link Pass 1 - Force Link Pass 0 - Do NOT Force Link Pass
		Write		
		Reset	1	
5	HD	**Read**		Duplex Mode 0 - Half Duplex 1 - Full Duplex NOTE: Valid only, if auto-negotiation for Duplex Mode is dis-abled.
		Write		
		Reset	0	

Table 30: General Purpose Control Register (GPCR)
Offset: 0x2804

Bits	Field		Type/ Reset Value	Description
4	FCTLRX	**Read**		Receive Flow Control Mode 0 - Enable IEEE 802.3x flow control 1 - Disable IEEE 802.3x flow control NOTE: Valid only, if auto-update for flow control is disabled.
		Write		
		Reset	0	
3	Speed	**Read**		Port Speed 0 - 10Mbps 1 - 100Mbps (and GigSpeed is set to 0) NOTE: Valid only, if Speed En bit is set.
		Write		
		Reset	1	
2	DPLYXen	**Read**		Enable Auto-Update for Duplex Mode 0 - Enable 1 - Disable
		Write		
		Reset	0	
1	FCTLen	**Read**		Enable Auto-Update for 802.3x Flow Control 0 - Enable 1 - Disable
		Write		
		Reset	1	
0	SpeedEn	**Read**		Enable Auto-Update for Speed 0 - Enable 1 - Disable
		Write		
		Reset	0	

Table 31: Transmit Control Register (TCR)
Offset: 0x2808

Bits	Field		Type/ Reset Value	Description
15	FJ	**Read**		Force Jam / Flow Control When in half-duplex mode, the CPU uses this bit to force collisions on the Ethernet segment. When the CPU recognizes that it is going to run out of receive buffers, it can force the transmitter to send jam frames, forcing collisions on the wire. The CPU must clear the FJ bit when more resources are available in order to allow transmission on the Ethernet segment. When in full-duplex mode and if Flow Control is enabled, this bit causes the port's transmitter to send Flow Control Pause packets. The CPU should reset this bit when more resources are available.
		Write		
		Reset	0	

Table 31: Transmit Control Register (TCR)
Offset: 0x2808

Bits	Field		Type/ Reset Value	Description
14	CRCD	**Read**		Insert CRC 0 - Enable 1 - Disable insertion of CRC in transmit packets When this bit is set, the MAC does not insert a CRC at the end of a transmit packet.
		Write		
		Reset	0	
13	PADD	**Read**		Pad Packets 0 - Enable 1 - Disable padding of packets of length less than 64 bytes When this bit is set, the MAC does not add padding to packets which are smaller than 64 bytes.
		Write		
		Reset	0	
12:10	ColTh	**Read**		Collision Threshold For Fast Ethernet: Number of TX clocks to count from the beginning of a packet before a collision is counted as a late collision. The number is in 32 cycle multiples (16 bytes transmit time). For Gigabit Ethernet: The number is fixed to 512 bytes.
		Write		
		Reset	100 (64 bytes)	
9:8	Reserved	**Read**	0	Reserved
		Write	0	
		Reset	0	
7:0	Padding Pattern	**Read**	00	When padding is enabled, these bits will allow to change padding patterns in 'byte'. For short packet padding. This pattern is programmable by this register.Value in this will be used as 'one byte' repetitions in the padding. Example, if you program 'AA', required 16 bytes padding will look like 'AA AA'.
		Write	00	
		Reset		

Table 32: Receive Control Register (RCR)
Offset: 0x280c

Bits	Field		Type/ Reset Value	Description
15	UnFiEn	**Read**		Unicast Filter Enable 0 - Disable 1 - Enable By setting this bit, the MAC will only pass packets with DA that matches either SA1 or SA2 on unicast packets. If only one address needs to be matched, the same address should be written in both SA1 and SA2.
		Write		
		Reset	0	
14	MuFiEn	**Read**		Multicast Filter Enable 0 - Disable 1 - Enable When this bit is set, MAC passes multicast packets to the DMA which have a DA. Produces a hit with the hash mechanism. MACAH1, MACAH2, MACAH3, and MACAH4 form the Hash Register. MACAH1 represents the least significant bits of Hash Register and MACAH4 the most significant bits. Possible modes: Bit 15 : Bit 14 0 0 - Promiscuous Mode 0 1 - Multicast Filtering Enabled 1 0 - Unicast Filtering Enabled 1 1 - Both Unicast and Multicast Filtering Enabled
		Write		
		Reset	0	
13	CRCR	**Read**		Remove CRC 0 - Keep CRC 1- Remove CRC Remove CRC from receive packets. The 4-byte CRC is removed from the received packets if this bit is set.
		Write		
		Reset	0	
12	PASSFC	**Read**		Pass Flow Control Packets 0 - Drop FC packets and do not send to FIFO 1 - Pass FC packets to FIFO NOTE: Only real packet is dropped, DA and SA of the FC packet are still passed in drop FC packets mode.
		Write		
		Reset		
11:0	Reserved	**Read**		Reserved
		Write		
		Reset	0	

Table 33: Transmit Flow Control Register (TFCR)
Offset: 0x2810

Bits	Field		Type/ Reset Value	Description
15:0	PauseTime	**Read**		Indicates the number of slot times during which the remote port receiving a flow-control packet from this port cannot send packets. This field is inserted into the transmitted flow control packets.
		Write		
		Reset	0xFFFF	

Table 34: Transmit Parameter Register (TPR)
Offset: 0x2814

Bits	Field		Type/ Reset Value	Description
15:14	JAM_Len	**Read**		Two bits to determine the JAM Length (in Backpressure) as follows: 00 = 12K bit times for FE 00 = 24K bit times for GIG 01 = 24K bit times for FE 01 = 48K bit times for GIG 10 = 32K bit times for FE 10 = 64K bit times for GIG 11 = 48K bit times for FE 11 = 96K bit times for GIG These values are only True when there is no data on the FIFO bus. If there is data, this data will be sent.
		Write		
		Reset	11 (48K bit time in FE mode 96K bit times in GIG mode)	
13:9	JAM_IPG	**Read**		Jam Inter Packet Gap For Fast Ethernet: 13:9 These bits determine the JAM IPG. The step is four bit times. The JAM IPG varies between 8-bit times and 124. For GIG: 11:9 = 011 010 - 32 bit times 011 - 48 bit times (default mode) 100 - 64 bit times 101 - 80 bit times 110 - 96 bit times 111 - 112 bit times NOTE: This value should be between above specified values only.
		Write		
		Reset	01011 (44 bit times in FE mode 48 bit times in GIG mode)	

Table 34: Transmit Parameter Register (TPR)
Offset: 0x2814

Bits	Field		Type/ Reset Value	Description
8:4	IPGJAM2Data	**Read**		Inter Packet Jam Data For Fast Ethernet: 8:4 These bits determine the IPG JAM to Data. The step is four bit times. The value varies between 8 bit times and 124. For GIG: 6:4 = 100 010 - 32 bit times 011 - 48 bit times 100 - 64 bit times (default mode) 101 - 80 bit times 110 - 96 bit times 111 - 112 bit times NOTE: This value should be between above specified values only.
		Write		
		Reset	11100 (112 bit times for FE) 64 bit times for GIG	
3:0	Back-off Limit	**Read**		These register bits set the back-off limit. These values are only valid if Limit4 in the SMR register is set.
		Write		
		Reset	0100	

Table 35: Serial Mode Register (SMR)
Offset: 0x2818

Bits	Field		Type/ Reset Value	Description
15:14	Reserved	**Read**		Reserved
		Write		
		Reset	0	
13:11	Data Blinder	**Read**		The number of nibbles from the beginning of the IPG, in which the IPG counter is restarted when detecting a carrier activity. Following this value, the port enters the Data Blinder zone and does not reset the IPG counter. This ensures fair access to the medium. Value should be written in Hex format. The step is 4-bit time. **NOTE:** These bits should only be changed at the start up or initialization and port is disabled.
		Write		
		Reset	00100	

Table 35: Serial Mode Register (SMR)
Offset: 0x2818

Bits	Field		Type/ Reset Value	Description
10	Limit4	**Read**		Limit4 The number of consecutive packet collisions that occurs before the collision counter is reset. 0 - The port resets its collision counter after 16 consecutive retransmit trials and restarts the Back off algorithm. 1- The port resets its collision counter and restarts the backoff algorithm after 4 consecutive transmit trials.
		Write		
		Reset	0	
9:8	Vlan_en, MFL	**Read**		VLAN Enabled, Maximum Frame Length 00 - Max. Frame Length = 1518 10 - Max. Frame Length = 1522 01 - Max. Frame Length = 9018 11 - Max. Frame Length = 9022
		Write		
		Reset	00	
7:5	Reserved	**Read**		Reserved
		Write		
		Reset	0	
4:0	IPGData	**Read**		Inter Packet Gap Data For Fast Ethernet: 4:0 Inter-Packet Gap (IPG): The step is 4 bit times. The value may vary between 48 bit times to 124. NOTE: These bits can only be changed when the Ethernet port is disabled. For GIG: 2:0
		Write		
		Reset	11000 (96 bit time)	

Table 36: Source Address Low (SAL1)
Offset: 0x281c

Bits	Field		Type/ Reset Value	Description
15:0	SA1 [15:0]	**Read**		Source Address The least significant bits of the source address for the port. This address is used for Flow Control.
		Write		
		Reset	0	

Table 37: Source Address Middle (SAM1)
Offset: 0x2820

Bits	Field		Type/ Reset Value	Description
15:0	SA1 [31:16]	**Read**		Source Address The middle bits of the source address for the port. This address is used for Flow Control.
		Write		
		Reset	0	

Table 38: Source Address High (SA1H)
Offset: 0x2824

Bits	Field		Type/ Reset Value	Description
15:0	SA1 [47:32]	**Read**		Source Address The most significant 16 bits of the source address for the port. This address is used for Flow Control.
		Write		
		Reset	0	

Table 39: Source Address Low (SAL2)
Offset: 0x2828

Bits	Field		Type/ Reset Value	Description
15:0	SA2 [15:0]	**Read**		Source Address Used for VLAN and others. The least significant bits of the source address for the port.
		Write		
		Reset	0	

Table 40: Source Address Middle (SAM2)
Offset: 0x282c

Bits	Field		Type/ Reset Value	Description
15:0	SA2 [31:16]	**Read**		Source Address Used for VLAN and others. The middle bits of the source address for the port.
		Write		
		Reset	0	

Table 41: Source Address High (SAH2)
Offset: 0x2830

Bits	Field		Type/ Reset Value	Description
15:0	SA 2[47:32]	**Read**		Source Address Used for VLAN and others. The most significant 16 bits of the source address for the port.
		Write		
		Reset	0	

Table 42: Multicast Address Hash Register 1 (MCAH1)
Offset: 0x2834

Bits	Field		Type/ Reset Value	Description
15:0	MCAH1[15:0]	**Read**		Multicast Address Hash Register 1
		Write		
		Reset	0	

Table 43: Multicast Address Hash Register 2 (MCAH2)
Offset: 0x2838

Bits	Field		Type/ Reset Value	Description
15:0	MCAH2[15:0]	**Read**		Multicast Address Hash Register 2
		Write		
		Reset	0	

Table 44: Multicast Address Hash Register 3 (MCAH3)
Offset: 0x283c

Bits	Field		Type/ Reset Value	Description
15:0	MCAH3[15:0]	**Read**		Multicast Address Hash Register 3
		Write		
		Reset	0	

Table 45: Multicast Address Hash Register 4 (MCAH4)
Offset: 0x2840

Bits	Field		Type/ Reset Value	Description
15:0	MCAH4[15:0]	**Read**		Multicast Address Hash Register 4
		Write		
		Reset	0	

Table 46: Transmit Interrupt Register (TIR)
Offset: 0x2844

Bits	Field		Type/ Reset Value	Description
15:0	TIR[15:0]	**Read**		Transmit Overflow Interrupt Register 0x70 to 0x8f of MIB counters.
		Write		
		Reset	0	
Transmit Interrupt Register Definitions				
15	Late	**Read**	Yes	The number of times a collision is detected later than 512-times into the transmission of a packet.
		Write	No	
		Reset		
14	Collisions	**Read**	Yes	The number of collisions experienced by a port during packet transmission.
		Write	No	
		Reset		

Table 46: Transmit Interrupt Register (TIR)
Offset: 0x2844

Bits	Field		Type/ Reset Value	Description
13	Spare	**Read**	Yes	
		Write	No	
		Reset		
12	OutMaxOctets	**Read**	Yes	The number of packets transmitted that were between 1519 and MAX_LEN bytes in length inclusive.
		Write	No	
		Reset		
11	Out1518Octets	**Read**	Yes	The number of packets transmitted that were between 1024 and 1518 bytes in length inclusive.
		Write	No	
		Reset		
10	Out1023Octets	**Read**	Yes	The number of packets transmitted that were between 512 and 1023 bytes in length inclusive.
		Write	No	
		Reset		
9	Out511Octets	**Read**	Yes	The number of packets transmitted that were between 256 and 511 bytes in length inclusive.
		Write	No	
		Reset		
8	Out255Octets	**Read**	Yes	The number of packets transmitted that were between 128 and 255 bytes in length inclusive.
		Write	No	
		Reset		
7	Out127Octets	**Read**	Yes	The number of packets transmitted that were between 65 and 127 bytes in length inclusive.
		Write	No	
		Reset		
6	Out64octets	**Read**	Yes	The number of packets transmitted that were 64 bytes in length.
		Write	No	
		Reset		
5:4	OutOctets	**Read**	Yes	The number of bytes of data transmitted, including bad packets. The count includes the FCS but not the preamble
		Write	No	
		Reset		

CONFIDENTIAL
Document Classification: Proprietary Information

Table 46: Transmit Interrupt Register (TIR)
Offset: 0x2844

Bits	Field		Type/ Reset Value	Description
3	OutMulticasts	**Read**	Yes	The number of Multicast packets transmitted by the port.
		Write	No	
		Reset		
2	OutPause	**Read**	Yes	The number of Pause packets transmitted by the port
		Write	No	
		Reset		
1	OutBroadcasts	**Read**	Yes	The number of Broadcast packets transmitted by the port.
		Write	No	
		Reset		
0	OutUnicasts	**Read**	Yes	The number of unicast packets transmitted by the port.
		Write	No	
		Reset		

Table 47: Receive Interrupt Register (RIR)
Offset: 0x2848

Bits	Field		Type/ Reset Value	Description
15:0	RIR[15:0]	**Read**		Receive Overflow Interrupt Register 0x40 to 0x5f of MIB counters
		Write		
		Reset	0	
Receive Interrupt Register Definitions				
15	In511Octets	**Read**	Yes	The number of packets (including bad packets) received that were between 256 and 511 bytes in length inclusive.
		Write	No	
		Reset		
14	In255Octets	**Read**	Yes	The number of packets (including bad packets) received that were between 128 and 255 bytes in length inclusive.
		Write	No	
		Reset		

Table 47: Receive Interrupt Register (RIR)
Offset: 0x2848

Bits	Field		Type/ Reset Value	Description
13	In127Octets	**Read**	Yes	The number of packets (including bad packets) received that were between 65 and 127 bytes in length inclusive.
		Write	No	
		Reset		
12	In64Octets	**Read**	Yes	The number of packets (including bad packets) received that were 64 bytes in length.
		Write	No	
		Reset		
11	Fragments	**Read**	Yes	The number of packets received by the port that are less than 64 bytes in long and have FCS error.
		Write	No	
		Reset		
10	Undersize	**Read**	Yes	The number of good packets received by the port that are less than 64 bytes long.
		Write	No	
		Reset		
9:8	InBadOctets	**Read**	Yes	The number of bytes of data received in bad packets. The count includes the FCS but not the preamble.
		Write	No	
		Reset		
7:6	InGoodOctets	**Read**	Yes	The number of good bytes of data received. The count includes the FCS but not the preamble.
		Write	No	
		Reset		
5	Spare	**Read**	Yes	
		Write	No	
		Reset		
4	InFCSErr	**Read**	Yes	The number of packets that have a bad FCS.
		Write	No	
		Reset		
3	InMulticasts	**Read**	Yes	The number of good Multicast packets received by the port.
		Write	No	
		Reset		

Table 47: Receive Interrupt Register (RIR)
Offset: 0x2848

Bits	Field		Type/ Reset Value	Description
2	InPause	**Read**	Yes	The number of PAUSE packets received by the port.
		Write	No	
		Reset		
1	InBroadcasts	**Read**	Yes	The number of good Broadcast packets received by the port.
		Write	No	
		Reset		
0	InUnicasts	**Read**	Yes	The number of good Unicast packets received by the port.
		Write	No	
		Reset		

Table 48: Transmit and Receive Interrupt Register (TIR_RIR)
Offset: 0x284c

Bits	Field		Type/ Reset Value	Description
15:0	TIR_RIR[15:0]	**Read**		7:0 - Receive Overflow Interrupt Register MSBs 0x60 to 0x6f of MIB counters 11:8 - Transmit Overflow Interrupt Register MSBs 0x90 to 0x97 of MIB counters
		Write		
		Reset	0	
Transmit and Receive Interrupt Register Definitions				
15:12	Reserved	**Read**		
		Write		
		Reset		
11	Underflow	**Read**	Yes	The number of Multicast packets transmitted by the port. The number of times a underflow condition occurs in the transmit FIFO.
		Write	No	
		Reset		
10	Single	**Read**	Yes	The number of successfully transmitted packets that experienced exactly one collision.
		Write	No	
		Reset		
9	Multiple	**Read**	Yes	The number of Broadcast packets transmitted by the port. The number of successfully transmitted packets that experienced more than one collision.
		Write	No	
		Reset		
8	Excessive	**Read**	Yes	The number of packets that are not transmitted from a port because the packet experienced 16 transmission attempts.
		Write	No	
		Reset		
7	Spare	**Read**	Yes	
		Write	No	
		Reset		
6	Overflow	**Read**	Yes	The number of times the overflow condition occurs.
		Write	No	
		Reset		
5	Spare	**Read**	Yes	
		Write	No	
		Reset		

Table 48: Transmit and Receive Interrupt Register (TIR_RIR)
Offset: 0x284c

Bits	Field		Type/ Reset Value	Description
4	Jabber	**Read**	Yes	The number of packets received that were longer than MAX_LEN bytes, and has a FCS error.
		Write	No	
		Reset		
3	OverSize	**Read**	Yes	The number of packets received that are longer than MAX_LEN bytes that were otherwise well formed.
		Write	No	
		Reset		
2	InMaxOctets	**Read**	Yes	The number of packets (including bad packets) received that were between 1519 and MAX_LEN bytes in length inclusive.
		Write	No	
		Reset		
1	In1518Octets	**Read**	Yes	The number of packets (including bad packets) received that were between 1024 and 1518 bytes in length inclusive.
		Write	No	
		Reset		
0	In1023Octets	**Read**	Yes	The number of packets (including bad packets) received that were between 512 and 1023 bytes in length inclusive.
		Write	No	
		Reset		

Table 49: Transmit Interrupt Mask Register (TIMR)
Offset: 0x2850

Bits	Field		Type/ Reset Value	Description
15:0	TIMR[15:0]	**Read**		Transmit Overflow Interrupt Register 0 - pass 1 - mask
		Write		
		Reset	0	

Table 50: Receive Interrupt Mask Register (RIMR)
Offset: 0x2854

Bits	Field		Type/ Reset Value	Description
15:0	RIMR[15:0]	**Read**		Receive Overflow Interrupt Register 0 - pass 1 - mask
		Write		
		Reset	0	

Table 51: Transmit and Receive Interrupt Mask Register (TIMR_RIMR)
Offset: 0x2858

Bits	Field		Type/ Reset Value	Description
15:0	TIMR_RIMR [15:0]	**Read**		7:0 - Receive Overflow Interrupt Register MSBs 11:8 Transmit Overflow MSBs 0 - pass 1 - mask
		Write		
		Reset		

Table 52: SMI Control Register (SMICR)
Offset: 0x2880

Bits	Field		Type/ Reset Value	Description
15:11	PhyAd	**Read**		PHY Device Address
		Write		
		Reset	0	
10:6	RegAd	**Read**		PHY Device Register Address
		Write		
		Reset	0	
5	OpCode	**Read**		OpCode 0 – Write 1 – Read
		Write		
		Reset	0	
4	ReadValid	**Read**		Read Valid Indicates that the Read operation has been completed for the addressed (RegAd) register, and the data is valid in the SMI Data Register. Read only.
		Write		
		Reset	0	
3	Busy	**Read**		Busy 0 - SMI interface is available 1 - Indicates that an operation is in progress and that CPU must not write to the SMI registers at this time. Read only.
		Write		
		Reset	0	
2:0	Reserved	**Read**		
		Write		
		Reset	0	

Table 53: SMI Data Register (SMIDR)
Offset: 0x2884

Bits	Field		Type/ Reset Value	Description
15:0	Data	**Read**		SMI Read Operation Two transactions are required: (1) Write to the SMI Control Register (SMICR) with OpCode = 1 and PhyAd, RegAd pointing to the PHY register to be read. (2) Read from the SMI Data Register (SMIDR). This read should be performed when the Read Valid bit in SMICR is set. The data remains undefined as long as Read Valid is 0. SMI WRITE operation Two transactions are required: (1) Write the register data to be written into the PHY register in the SMI Data Register (SMIDR). (2) Write to the SMI Control Register (SMICR) with OpCode = 0 and PhyAd, RegAd pointing to the PHY register to be written to.
		Write		
		Reset	0	

Table 54: PHY Address Register (PAR)
Offset: 0x2888

Bits	Field		Type/ Reset Value	Description
15:6	Reserved	**Read**		
		Write		
		Reset	0	
5	MIBclrMode	**Read**		MIB Counters Clear Mode Setting this bit causes the counters to reset when the CPU reads a counter. In order to reset all MIB counters, the CPU should set this bit and read all the counters individually. The reset is only performed when the lower 16 bits of the counters are read and if MIBclrMode is set.
		Write		
		Reset	0	
4	LoadTstCnt	**Read**		Load a count of FFFF_FFF0 into the RMON counters when it is read. This is used only in test mode.
		Write		
		Reset	000	
3:0	Reserved	**Read**		
		Write		
		Reset		

MIB Counters

Address in Control Register File	Field	Bits	Description
0x2900	InUnicasts	15:0	The number of good Unicast packets received by the port.
0x2904	InUnicasts	31:16	
0x2908	InBroadcasts	15:0	The number of good broadcast packets received by the port.
0x290c	InBroadcasts	31:16	
0x2910	In Pause	15:0	The number of Pause packets received by the port.
0x2914	InPause	31:16	
0x2918	InMulticasts	15:0	The number of good Multicast packets received by the port.
0x291c	InMulticasts	31:16	
0x2920	InFCSErr	15:0	The number of packets that have a bad FCS.
0x2924	InFCSErr	31:16	
0x2928	Spare		
0x292c	Spare		
0x2930	InGoodOctets	15:0	The number of good bytes of data received. The count includes the FCS but not the preamble.
0x2934	InGoodOctets	31:16	
0x2938	InGoodOctets	47:32	
0x293c	InGoodOctets	63:48	
0x2940	InBadOctets	15:0	The number of bytes of data received in bad packets. The count includes the FCS but not the preamble.
0x2944	InBadOctets	31:16	
0x2948	InBadOctets	47:32	
0x294c	InBadOctets	63:48	
0x2950	Undersize	15:0	The number of good packets received by the port are less than 64 bytes long.
0x2954	Undersize	31:16	
0x2958	Fragments	15:0	The number of packets received by the port that are less than 64 bytes long and have a FCS error.
0x295c	Fragments	31:16	

Address in Control Register File	Field	Bits	Description
0x2960	In64Octets	15:0	The number of packets (including bad packets) received that were 64 bytes in length.
0x2964	In64Octets	31:16	
0x2968	In127Octets	15:0	The number of packets (including bad packets) received that were between 65 and 127 bytes in length.
0x296c	In127Octets	31:16	
0x2970	In255Octets	15:0	The number of packets (including bad packets) received that were between 128 and 255 bytes in length.
0x2974	In255Octets	31:16	
0x2978	In511Octets	15:0	The number of packets (including bad packets) received that were between 256 and 511 bytes in length.
0x297c	In511Octets	31:16	
0x2980	In1023Octets	15:0	The number of packets (including bad packets) received that were between 512 and 1023 bytes in length.
0x2984	In1023Octets	31:16	
0x2988	In1518Octets	15:0	The number of packets (including bad packets) received that were between 1024 and 1518 bytes in length.
0x298c	In 1518Octets	31:16	
0x2990	InMaxOctets	15:0	The number of packets (including bad packets) received that were between 1519 and MAX_LEN bytes in length.
0x2994	InMaxOctets	31:16	
0x2998	OverSize	15:0	The number of packets received that are longer than MAX_LEN bytes and were well formed.
0x299c	Oversize	31:16	
0x29a0	Jabber	15:0	The number of packets received that were longer than MAX_LEN and had an FCS error.
0x29a4	Jabber	31:16	
0x29a8	Spare		
0x29ac	Spare		
0x29b0	Overflow	15:0	The number of times the overflow condition occurs.
0x29b4	Overflow	31:16	
0x29b8	Spare		
0x29bc	Spare		

Address in Control Register File	Field	Bits	Description
0x29c0	OutUnicasts	15:0	The number of Unicast packets transmitted by the port.
0x29c4	OutUnicasts	31:16	
0x29c8	OutBroadcasts	15:0	The number of Broadcast packets transmitted by the port.
0x29cc	OutBroadcasts	31:16	
0x29d0	OutPause	15:0	The number of Pause Packets transmitted by the port.
0x29d4	OutPause	31:16	
0x29d8	OutMulticasts	15:0	The number of good Multicast packets transmitted by the port.
0x29dc	OutMulticasts	31:16	
0x29e0	OutOctets	15:0	The number of bytes of data transmitted, including bad packets. The count includes FCS but not the preamble.
0x29e4	OutOctets	31:16	
0x29e8	OutOctets	47:32	
0x29ec	OutOctets	63:48	
0x29f0	Out64Octets	15:0	The number of packets (including bad packets) transmitted that were 64 bytes in length.
0x29f4	Out64Octets	31:16	
0x29f8	Out127Octets	15:0	The number of packets (including bad packets) transmitted that were between 65 and 127 bytes in length.
0x29fc	Out127Octets	31:16	
0x2a00	Out255Octets	15:0	The number of packets (including bad packets) transmitted that were between 128 and 255 bytes in length.
0x2a04	Out255Octets	31:16	
0x2a08	Out511Octets	15:0	The number of packets (including bad packets) transmitted that were between 256 and 511 bytes in length.
0x2a0c	Out511Octets	31:16	
0x2a10	Out1023Octets	15:0	The number of packets (including bad packets) transmitted that were between 512 and 1023 bytes in length.
0x2a14	Out1023Octets	31:16	
0x2a18	Out1518Octets	15:0	The number of packets (including bad packets) transmitted that were between 1024 and 1518 bytes in length.
0x2a1c	Out 1518Octets	31:16	

Address in Control Register File	Field	Bits	Description
0x2a20	OutMaxOctets	15:0	The number of packets (including bad packets) transmitted that were between 1519 and MAX_LEN bytes in length.
0x2a24	OutMaxOctets	31:16	
0x2a28	Spare		
0x2a2c	Spare		
0x2a30	Collisions	15:0	The number of collisions experienced by a port during packet transmission.
0x2a34	Collisions	31:16	
0x2a38	Late	15:0	The number of times that a collision is detected later than 512 bit-times into the transmission of a packet.
0x2a3c	Late	31:16	
0x2a40	Excessive	15:0	The number of packets that are not transmitted from a port because the packet experienced 16 transmission attempts.
0x2a44	Excessive	31:16	
0x2a48	Multiple	15:0	The number of successfully transmitted packets that experienced more than one collision.
0x2a4c	Multiple	31:16	
0x2a50	Single	15:0	The number of successfully transmitted packets that experienced exactly one collision.
0x2a54	Single	31:16	
0x2a58	Underflow	15:0	The number of times an underflow condition occurs in the transmit FIFO.
0x2a5c	Underflow	31:16	

Section 4. Electrical Specifications

4.1 Absolute Maximum Ratings

Stresses above those listed in Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

Symbol	Parameter	Min	Typ	Max	Units
$V_{DD(3.3)}$	Power Supply Voltage on A_{VDDH} with respect to V_{SS}	-0.5		+3.6	V
$V_{DD(2.5)}$	Power Supply Voltage on V_{DDO_TTL} with respect to VSS	-0.5		+3.6 or $V_{DD(3.3)}$ +0.5[1] whichever is less	V
$V_{DD(1.2)}$	Power Supply Voltage on V_{DD} with respect to V_{SS}	-0.5		+3.6 or $V_{DD(2.5)}$ +0.5[2] whichever is less	V
V_{PIN}	Voltage applied to any input pin with respect to V_{SS}	-0.5		+3.6 or V_{DDO} +0.5[3] whichever is less	V
$T_{STORAGE}$	Storage temperature	-55		+125[4]	°C

1. VDD(2.5) must never be more than 0.5V greater than VDD(3.3) or damage will result. This implies that power must be applied to VDD(3.3) before or at the same time as VDD(2.5).
2. VDD(1.2) must never be more than 0.5V greater than VDD(2.5) or damage will result. This implies that power must be applied to VDD(2.5) before or at the same time as VDD(1.2).
3. VPIN must never be more than 0.5V greater than VDDO or damage will result.
4. 125°C is the re-bake temperature. For extended storage time greater than 24 hours, +85°C should be the maximum.

4.2 Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Units
$V_{DD(3.3)}$	3.3V power supply	For pins V_{DDO_TTL}	3.135	3.3	3.465	V
$V_{DD(2.5)}$	2.5V power supply	For pins A_{VDDL}	2.375	2.5	2.625	V
$V_{DD(1.2)}$	1.2V power supply	For pins V_{DD}	1.140	1.2	1.260	V
T_A	Ambient operating temperature		0		70	°C
T_J	Maximum junction temperature				125[1]	°C
RSET	Internal bias reference	Constant voltage reference. External 4.99 kΩ 1% resistor connection to VSS.	2465	2490	2515	Ω

1. Refer to white paper on TJ Thermal Calculations for more Information.

4.3 Package Thermal Information

4.3.1 Thermal Conditions for 64-pin QFN Package

Symbol	Parameter	Condition	Min	Typ	Max	Units
θ_{JA}	Thermal resistance[1] - junction to ambient of the 88E8053 device 64-Pin QFN package $\theta_{JA} = (T_J - T_A)/ P$ P = Total Power Dissipation	JEDEC 3 in. x 4.5 in. 4-layer PCB with no air flow		23.30		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 1 meter/sec air flow		20.60		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 2 meter/sec air flow		19.60		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 3 meter/sec air flow		19.00		°C/W
ψ_{JT}	Thermal characteristic parameter[1] - junction to top center of the 88E8053 device 64-Pin QFN package $\psi_{JT} = (T_J - T_{TOP})/P$. Ttop = Temperature on the top center of the package	JEDEC 3 in. x 4.5 in. 4-layer PCB with no air flow		0.17		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 1 meter/sec air flow		0.40		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 2 meter/sec air flow		0.50		°C/W
		JEDEC 3 in. x 4.5 in. 4-layer PCB with 3 meter/sec air flow		0.58		°C/W
θ_{JC}	Thermal resistance[1] - junction to case of the 88E8053 device 64-Pin QFN package $\theta_{JC} = (T_J - T_C)/ P_{Top}$ P_{Top} = Power Dissipation from the top of the package	JEDEC with no air flow		9.30		°C/W
θ_{JB}	Thermal resistance[1] - junction to board of the 88E8053 device 64-Pin QFN package $\theta_{JB} = (T_J - T_B)/ P_{bottom}$ P_{bottom} = power dissipation from the bottom of the package to the PCB surface.	JEDEC with no air flow		15.50		°C/W

1. Refer to white paper on TJ Thermal Calculations for more information.

4.4 DC Electrical Characteristics

4.4.1 Current Consumption AVDDL

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
I_{AVDDL}	2.5V Power to analog core - copper	AVDDL	No Link		82		mA
			1000 Mbps Traffic		218		mA
			100 Mbps Traffic		126		mA
			10 Mbps Traffic		108		mA
			LOM_DISABLE without Switch		4		mA
			D3 Cold with PME disabled without Switch		12		mA
			D3 Cold without PME disabled without switch		27		mA

4.4.2 Current Consumption VDD

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
I_{VDD}	Core power (1.2V)	VDD	No Link		171		mA
			1000 Mbps Traffic		426		mA
			100 Mbps Traffic		203		mA
			10 Mbps Traffic		179		mA
			LOM_DISABLE without Switch		13		mA
			D3 Cold with PME disabled without Switch		29		mA
			D3 Cold without PME disabled without switch		130		mA

4.4.3 Current Consumption VDDO_TTL

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
I_{VDDO_TTL}	PCI I/O power (3.3V)	VDDO_TTL	No Link		4		mA
			1000 Mbps Traffic		4		mA
			100 Mbps Traffic		4		mA
			10 Mbps Traffic		4		mA
			LOM_DISABLE without Switch		3		mA
			D3 Cold with PME disabled without Switch		3		mA
			D3 Cold without PME disabled without switch		3		mA

4.4.4 Digital Operating Conditions

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
V_{IH}	High level input voltage	All pins		2.0		V_{DD} +0.5	V
		XTALI					
V_{IL}	Low level input voltage	All pins		-0.5		0.8	V
		XTALI					
V_{OH}	High level output voltage	LED pins[1]		2.4			V
		XTALO					V
		All others (except INTAn)		2.4			V
V_{OL}	Low level output voltage	LED pins				0.4	V
		XTALO					V
		INTAn pin					V
		All others				0.4	V
I_{ILK}	Input leakage current	With pull-up resistor	$0<V_{IN}<V_{DD}$				µA
		With pull-down resistor	$0<V_{IN}<V_{DD}$				µA
		All others	$0<V_{IN}<V_{DD}$				µA
C_{IN}	Input capacitance	All pins				5	pF

1. The LED pins are as follows: LED_ACTn, LED_LINK10/100n, LED_LINK1000n, LED_LINKn.

Table 55: 88E8053 Internal Resistor Description

Pin #	Pin Name	Resistor	Pin #	Pin Name	Resistor
38	VPD_CLK	Internal pull-up	37	SPI_CLK	Internal pull-up
41	VPD_DATA	Internal pull-up	36	SPI_CS	Internal pull-up
34	SPI_DO	Internal pull-up	46	TESTMODE	Default pull-down
35	SPI_DI	Internal pull-up			

4.4.5 IEEE DC Transceiver Parameters

IEEE tests are typically based on templates and cannot simply be specified by a number. For an exact description of the template and the test conditions, refer to the IEEE specifications:
-10BASE-T IEEE 802.3 Clause 14
-100BASE-TX ANSI X3.263-1995
(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
V_{ODIFF}	Absolute peak differential output voltage	MDI[1:0]	10BASE-T no cable	2.2	2.5	2.8	V
		MDI[1:0]	10BASE-T cable model	585[1]			mV
		MDI[1:0]	100BASE-TX mode	0.950	1.0	1.050	V
		MDI[3:0]	1000BASE-T[2]	0.67	0.75	0.82	V
	Overshoot[2]	MDI[1:0]	100BASE-TX mode	0		5%	V
	Amplitude Symmetry (positive/ negative)	MDI[1:0]	100BASE-TX mode	0.98x		1.02x	V+/V-
V_{IDIFF}	Peak Differential Input Voltage	MDI[1:0]	10BASE-T mode	585[3]			mV
	Signal Detect Assertion	MDI[1:0]	100BASE-TX mode	1000	460[4]		mV peak-peak
	Signal Detect De-assertion	MDI[1:0]	100BASE-TX mode	200	360[5]		mV peak-peak

1. IEEE 802.3 Clause 14, Figure 14.9 shows the template for the “far end” wave form. This template allows as little as 495 mV peak differential voltage at the far end receiver.
2. IEEE 802.3ab Figure 40 -19 points A&B.
3. The input test is actually a template test ; IEEE 802.3 Clause 14, Figure 14.17 shows the template for the receive wave form.
4. The ANSI TP-PMD specification requires that any received signal with peak-to-peak differential amplitude greater than 1000 mV should turn on signal detect (internal signal in 100BASE-TX mode). The 88E8053 will accept signals typically with 460 mV peak-to-peak differential amplitude.
5. The ANSI TP-PMD specification requires that any received signal with peak-to-peak differential amplitude less than 200 mV should de-assert signal detect (internal signal in 100BASE-TX mode). The 88E8053 will reject signals typically with peak-to-peak differential amplitude less than 360 mV.

4.5 AC Timing Reference Values

Symbol	Parameter	Pins	Min	Typ	Max	Units
V_{IH} (Min.)	Input high voltage reference		1.9			V
V_{IL} (Max.)	Input low voltage reference				0.7	V
V_{OH} (Min.)	Output high voltage reference		1.9			V
V_{OL} (Max.)	Output low voltage reference				0.7	V

4.6 AC Electrical Specifications

4.6.1 Reset Timing

(Over Full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Condition	Min	Typ	Max	Units
T1	100 ms from PCI-Express Spec.					
T2	1ms < T2 < 12 ms					
T3	150 ms					
T4	5 ms					
T5	150 ms					

Figure 15: Timing Requirements from BIOS (LOM_DISABLE starts with low)

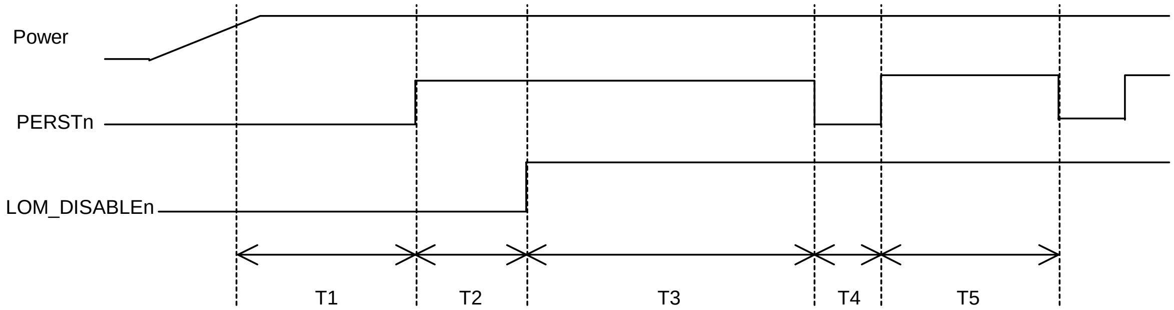
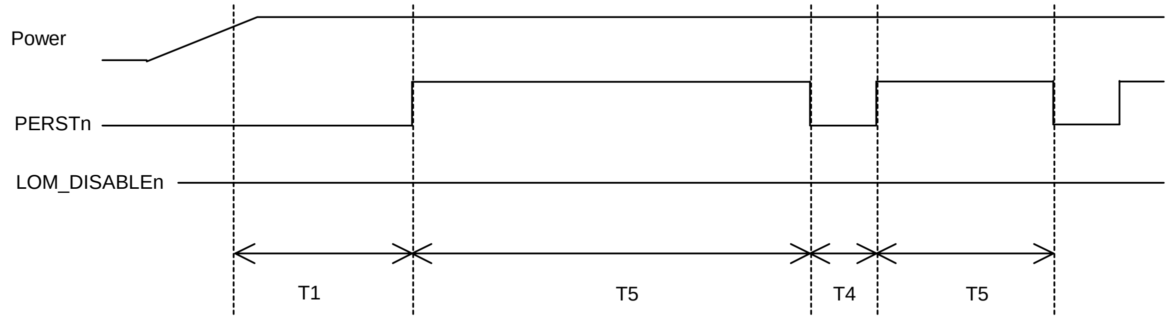


Figure 16: Timing Requirements from BIOS (LOM_DISABLE starts with high)



4.6.2 Device Wakeup Timing

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Condition	Min	Typ	Max	Units
T1	PERSTn from LOM_DISABLEn			150		ms
T2	PERSTn Timing			5		ms

Figure 17: Device Enable from LOM Disable State

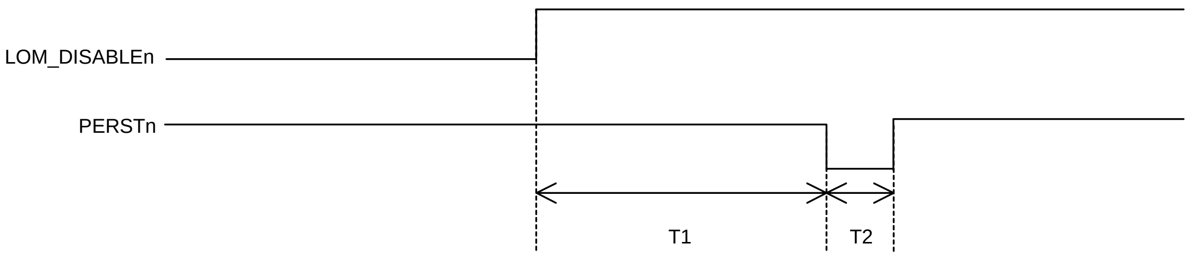
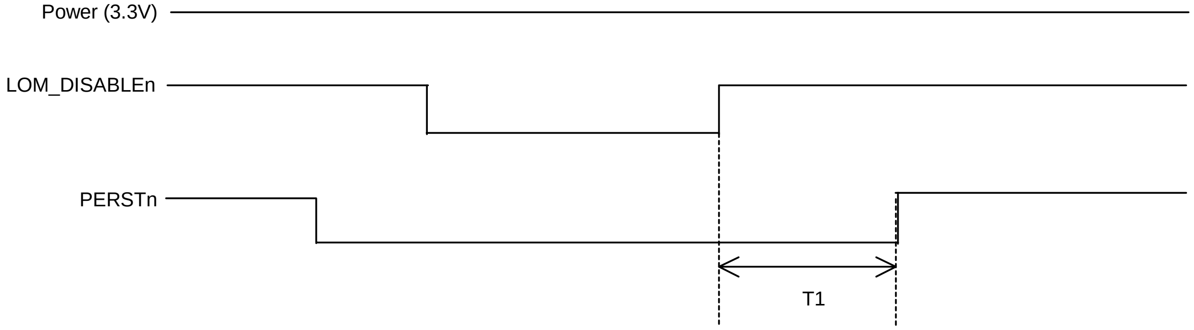


Figure 18: Device Wakeup from Power Management State (D3 Cold)



CONFIDENTIAL
Document Classification: Proprietary Information

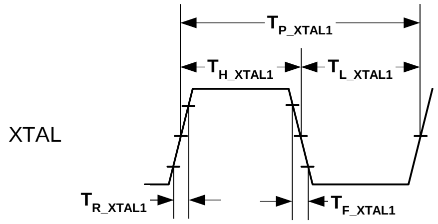
April 20, 2004, Advanced

4.6.3 Clock Timing

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Condition	Min	Typ	Max	Units
T_{P_XTAL}	XTAL Period	±50 ppm	40	40	40	ns
T_{H_XTAL}	XTAL High time		13	20	27	ns
T_{L_XTAL}	XTAL Low time		13	20	27	ns
T_{R_XTAL}	XTAL Rise	10% to 90%	-	-	3.0	ns
T_{F_XTAL}	XTAL Fall	90% to 10%	-	-	3.0	ns

Figure 19: Clock Timing



4.6.4 PCI Express Timing

4.6.4.1 Differential Transmitter (TX) Output Specifications

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Min	Typ	Max	Units
UI	Unit Interval Each UI is 400 ps ±300 ppm. UI does not account for SSC dictated variations.	399.88	400	400.12	ps
$V_{TX\_DIFFp\text{-}p}$	Differential Peak to Peak Output Voltage $V_{TX\_DIFFp\text{-}p}$=2*I$V_{TX\text{-}D+}$ - $V_{TX\text{-}D\text{-}}$I	0.800		1.2	V
$V_{TX\_DE\_RATIO}$	De-Emphasized Differential Output Voltage (Ratio)	-3.0	-3.5	-4.0	dB
$T_{TX\text{-}EYE}$	Minimum TX Eye Width	0.70			UI
$T_{TX\text{-}EYE\text{-}MEDIAN\text{-}to\text{-}MAX\text{-}JITTER}$	Maximum time between the jitter median and maximum deviation from the median			0.15	UI
$T_{TX\text{-}RISE}$, $T_{TX\text{-}FALL}$	D+/D- TX Output Rise/Fall Time	0.125			UI
$V_{TX\text{-}CM\text{-}ACp}$	RMS AC Peak Common MOde Output Voltage			20	mV
$V_{TX\text{-}CH\text{-}DC\text{-}ACTIVE\text{-}IDLE\text{-}DELTA}$	Absolute Delta Common Mode Voltage During L0 and Electrical Idle	0		100	mV
$V_{TX\text{-}CM\text{-}DC\text{-}LINE\text{-}DELTA}$	Absolute Delta of DC Common Mode Voltage between D+ and D-	0		25	mV
$V_{TX\text{-}IDLE\text{-}DIFFp}$	Electrical Idle Differential Peak Output Voltage	0		20	mV
$V_{TX\text{-}RCV\text{-}DETECT}$	The amount of voltage change allowed during Receiver Detection			600	mV
$V_{TX\text{-}DC\text{-}CM}$	The TX DC Common Mode Voltage	0		3.6	V
$I_{TX\text{-}SHORT}$	TX Short Circuit Current Limit			90	mA
$T_{TX\text{-}IDLE\text{-}MIN}$	Minimum time spent in Electrical Idle	50			UI
$T_{TX\text{-}IDLE\text{-}SET\text{-}TO\text{-}IDLE}$	Maximum time to transition to a valid Electrical Idle after sending an Electrical Idle ordered set			20	UI
$T_{TX\text{-}IDLE\text{-}TO\text{-}TO\text{-}DIFF\text{-}DATA}$	Maximum time to transition to valid TX specifications after leaving an Electrical Idle condition			20	UI
$RL_{TX\text{-}DIFF}$	Differential Return Loss	12			dB
$RL_{TX\text{-}CM}$	Common Mode Return Loss	6			dB
$Z_{TX\text{-}DIFF\text{-}DC}$	DC Differential TX Impedance	80	100	120	Ω
$Z_{TX\text{-}DC}$	Transmitter DC Impedance	40			Ω
C_{TX}	AC Coupling Capacitor	75		200	nF
$T_{crosslink}$	Crosslink Random Timeout	0		1	ms

4.6.4.2 Differential Receiver (RX) Output Specifications

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

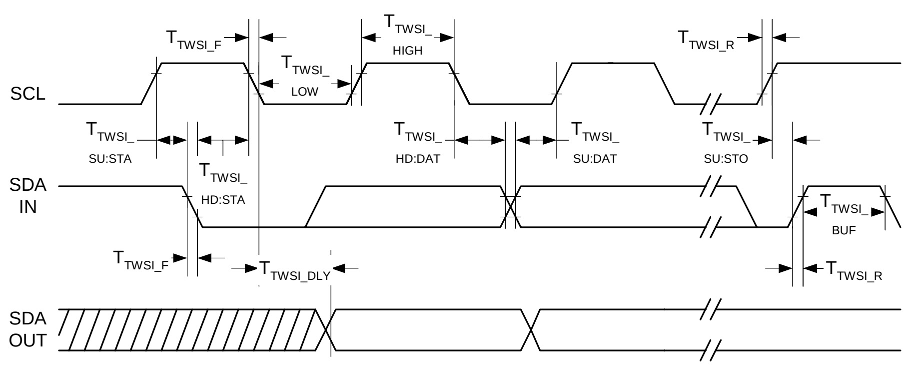
Symbol	Parameter	Min	Typ	Max	Units
UI	Unit Interval	399.88	400	400.12	ps
$V_{TX\_DIFFp-p}$	Differential Peak to Peak Output Voltage $V_{TX\_DIFFp-p}=2*\lvert V_{TX-D+} - V_{TX-D-}\rvert$	0.175		1.2	V
T_{RX-EYE}	Minimum RX Eye Width	0.4			UI
$T_{RX-EYE-MEDIAN-to-MAX-JITTER}$	Maximum time between the jitter median and maximum deviation from the median			0.3	UI
$V_{RX-CM-ACp}$	AC Peak Common MOde Input Voltage			150	mV
$RL_{RX-DIFF}$	Differential Return Loss	15			dB
RL_{RX-CM}	TCommon Mode Return Loss	0		3.6	dB
$Z_{RX-DIFF-DC}$	DC Differential Input Impedance	80	100	120	Ω
Z_{RX-DC}	DC Input Impedance	40	50	60	Ω
$Z_{RX-HIGH-IMP-DC}$	Powered Down DC Input Impedance	200 k			Ω
$V_{RX-IDLE-DET-DIFF-p-p}$	Electrical Idle Detect Threshold	65		175	mV
$T_{RX-IDLE-DET-DIFF-ENTERTIME}$	Unexpected Electrical Idle Enter Detect Threshold Integration Time			10	ms
$L_{RX-SKEW}$	Total Skew			20	ns

4.6.5 Two-Wire Serial Interface (TWSI) Timing

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Condition	Min	Typ	Max	Units
F_{TWSI_SCL}	SCL Clock Frequency	100 kHz			100	kHz
		400 kHz			400	
T_{TWSI_NS}	Noise Suppression Time at SCL, SDA Inputs	100 kHz			80	ns
		400 kHz			80	
T_{TWSI_R}	SDA Rise time	100 kHz			1000	ns
		400 kHz			300	
T_{TWSI_F}	SDA Fall Time	100 kHz			300	ns
		400 kHz			300	
T_{TWSI_HIGH}	Clock High Period	100 kHz	4000			ns
		400 kHz	600			
T_{TWSI_LOW}	Clock Low Period	100 kHz	4700			ns
		400 kHz	1300			
$T_{TWSI\_SU:STA}$	Start Condition Setup Time (for a Repeated Start Condition)	100 kHz	4700			ns
		400 kHz	600			
$T_{TWSI\_HD:STA}$	Start Condition Hold Time	100 kHz	4000			ns
		400 kHz	600			
$T_{TWSI\_SU:STO}$	Stop Condition Setup Time	100 kHz	4000			ns
		400 kHz	600			
$T_{TWSI\_SU:DAT}$	Data in Setup Time	100 kHz	250			ns
		400 kHz	100			
$T_{TWSI\_HD:DAT}$	Data in Hold Time	100 kHz	0			ns
		400 kHz	0			
T_{TWSI_BUF}	Bus Free Time	100 kHz	4700			ns
		400 kHz	1300			
T_{TWSI_DLY}	SCL Low to SDA Data Out Valid	100 kHz	40		200	ns
		400 kHz	40		200	

Figure 20: Two-Wire Serial Interface Timing



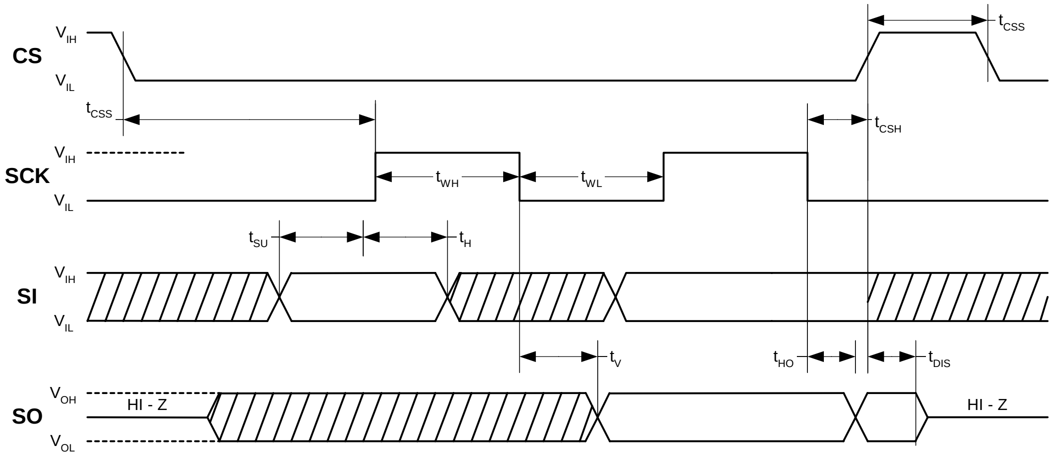
4.6.6 SPI FLash Memory Interface Timing

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Min	Typ	Max	Units
F_{SCK}	SCK Clock Frequency	0		20	MHz
t_{RI}	Input Rise Time			20	ns
t_{FI}	Input Fall Time			20	ns
t_{WH}	SCK High Time	20			ns
t_{WL}	SCK Low Time	20			ns
t_{CS}	CS High Time	25			ns
t_{CSS}	CS Setup Time	25			ns
t_{CSH}	CS Hold Time	25			ns
t_{SU}	Data In Setup Time	5			ns
t_{H}	Data In Hold Time	5			ns
t_{HD}	Hold Setup Time	15			ns
t_{CD}	Hold Time	15			ns
t_{V}	Output Valid			20	ns
t_{HO}	Output Hold Time	0			ns
t_{LZ}	Hold to Output Low Z			200	ns
t_{HZ}	Hold to Output High Z			200	ns
t_{DIS}	Output Disable Time			100	ns
t_{EC}	Erase Cycle Time per Sector			1.1	s
t_{BPC}	Byte Program Cycle Time[1]		60	100	µs
Endurance[2]			10K		Write Cycles[3]

1. The programming time for n bytes will be equal to n * tBPC.
2. This parameter is characterized at 3.0V, 25°C and is not 100% tested.
3. One write cycle consists of erasing a sector, followed by programming the same sector.

Figure 21: Synchronous Data Timing

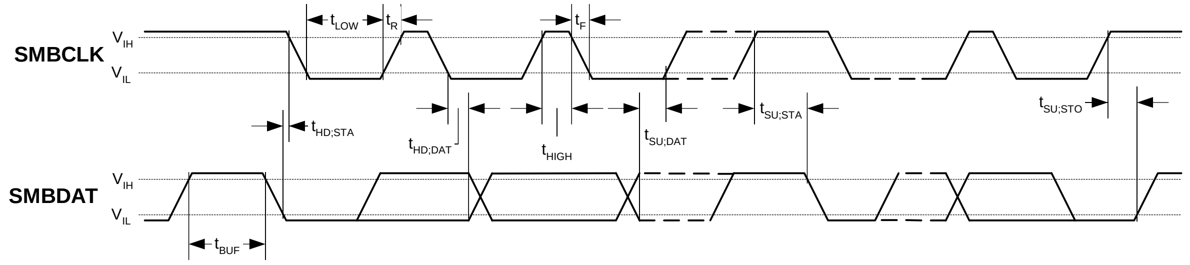


4.6.7 SMBUS Specifications

For an exact description of the SMBus 2.0 Electrical characteristics, refer to the SMBus specification:
-System Management Bus (SMBus) Specification Version 2.0
(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

Symbol	Parameter	Min	Typ	Max	Units
F_{SMB}	SMBus Operating Frequency	10		100	KHz
T_{BUF}	Bus free time between stop and Start Condition	4.7			µs
$T_{HD;STA}$	Hold time after (Repeated) Start Condition. After this period, the first clock is generated.	4.0			µs
$T_{SU;STA}$	Repeated Start Condition Setup	4.7			µs
$T_{SU;STO}$	Stop condition after setup time	4.0			µs
$T_{HD;DAT}$	Data hold time	300			ns
$T_{SU;DAT}$	Data setup time	250			ns
$T_{TIMEOUT}$	Detect clock low timeout	25		35	ms
T_{LOW}	Clock Low Period	4.7			µs
T_{HIGH}	Clock High Period	4.0		50	µs
$T_{LOW;SEXT}$	Cumulative clock low extend time (slave device)			25	ms
$T_{LOW;MEXT}$	Cumulative clock low extend time (master device)			10	ms
T_F	Clock/Data Fall Time			300	ns
T_R	Clock/Data Rise Time			1000	ns
T_{POR}	Time in which a device must be operational after power-on reset			500	ms

Figure 22: SMBus AC Specifications



4.7 IEEE AC Parameters

IEEE tests are typically based on templates and cannot simply be specified by number. For an exact description of the templates and the test conditions, refer to the IEEE specifications:
-10BASE-T IEEE 802.3 Clause 14-2000
-100BASE-TX ANSI X3.263-1995
-1000BASE-T IEEE 802.3ab Clause 40 Section 40.6.1.2 Figure 40-26 shows the template waveforms for transmitter electrical specifications.

(Over full range of values listed in the Recommended Operating Conditions unless otherwise specified)

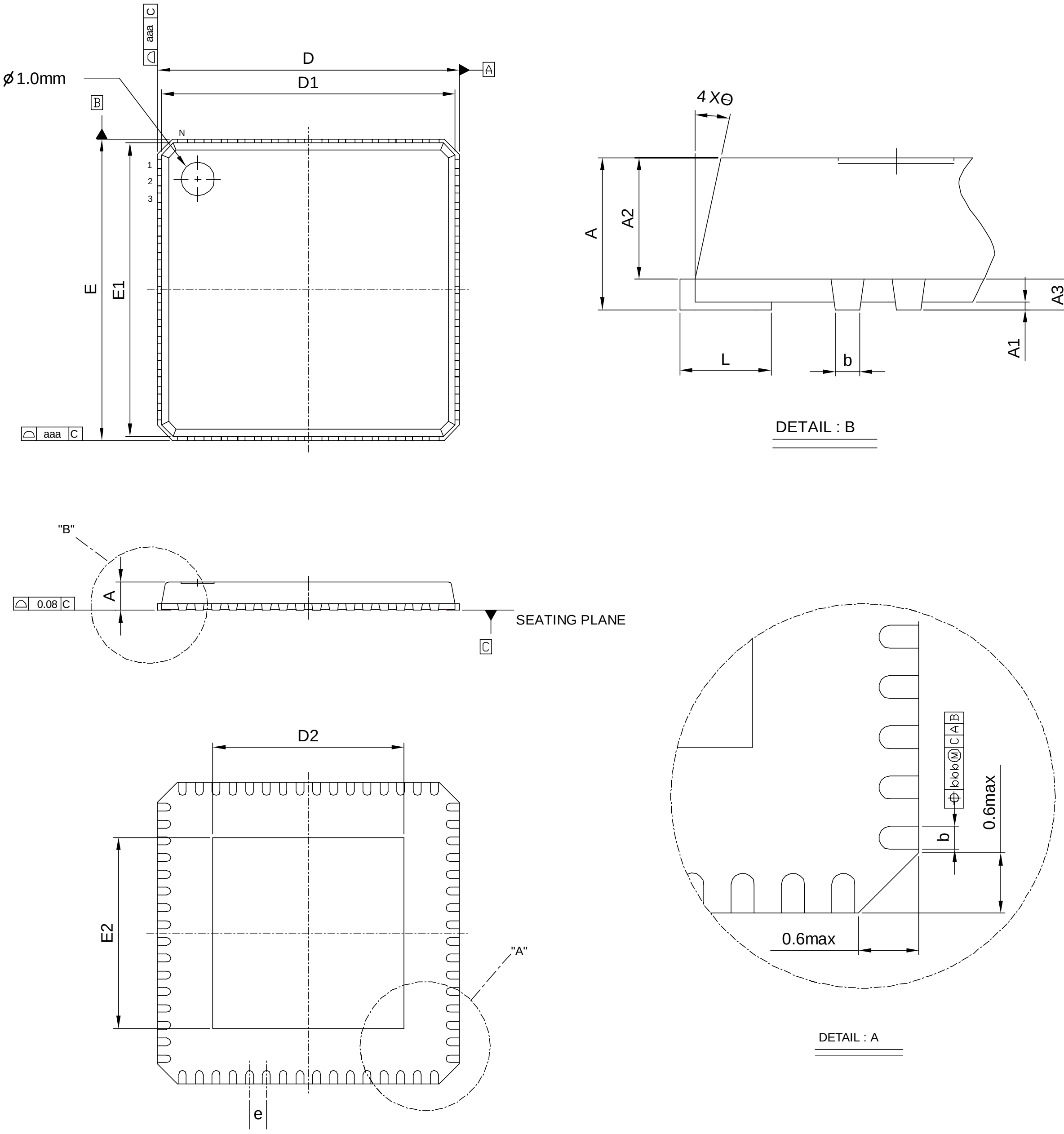
Symbol	Parameter	Pins	Condition	Min	Typ	Max	Units
T_{RISE}	Rise time	MDI[1:0]	100BASE-TX	3.0	4.0	5.0	ns
T_{FALL}	Fall time	MDI[1:0]	100BASE-TX	3.0	4.0	5.0	ns
T_{RISE}/ T_{FALL} Symmetry		MDI[1:0]	100BASE-TX	0		0.5	ns
DCD	Duty cycle distortion	MDI[1:0]	100BASE-TX	0		0.5[1]	ns, peak-peak
Transmit Jitter		MDI[1:0]	100BASE-TX	0		1.4	ns, peak-peak

1. ANSI X3.263-1995 Figure 9-3.

Section 5. Mechanical Drawings

5.1 64-Pin QFN Package

Figure 23: 88E8053 64-Pin QFN Package



(All dimensions in mm)

Table 56: 64-Pin QFN Mechanical Dimensions

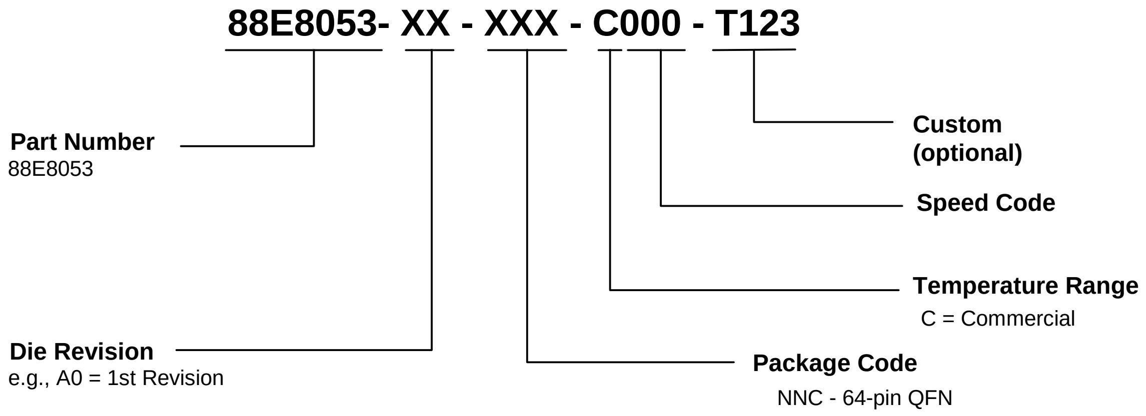
	Dimensions in mm		
Symbol	**MIN**	**NOM**	**MAX**
A	0.80	0.85	1.00
A1	0.00	0.02	0.05
A2	--	0.65	1.00
A3	0.20 REF		
b	0.18	0.23	0.30
D	9.00 BSC		
D1	8.75 BSC		
E	9.00 BSC		
E1	8.75 BSC		
e	0.50 BSC		
L	0.30	0.40	0.50
θ	0°	--	12°
aaa	--	--	0.25
bbb	--	--	0.10
chamfer	--	--	0.60

Die Pad Size	
Symbol	**Dimension in mm**
D_2	5.46 ± 0.20
E_2	6.25 ± 0.20

Section 6. Order Information

Figure 24 shows the ordering part numbering scheme for the 88E8053. Contact Marvell® or sales representatives for complete ordering information.

Figure 24: Sample Ordering Part Number

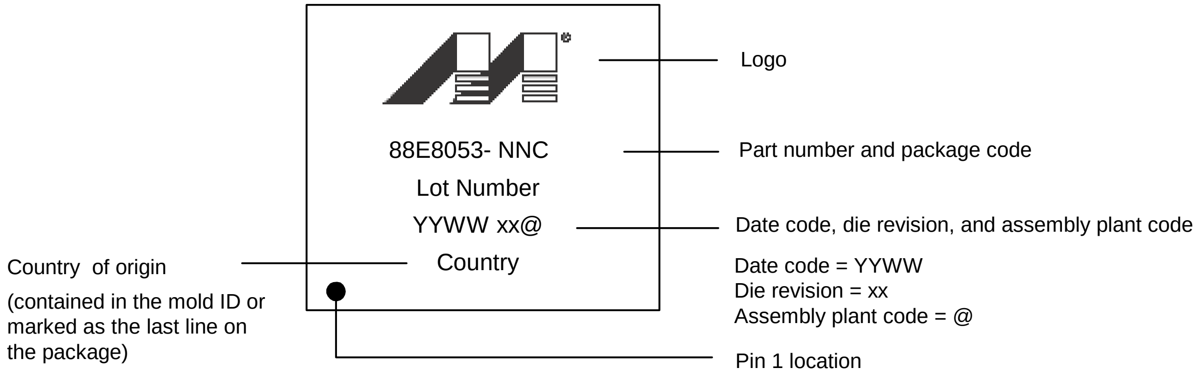


The standard ordering part number is:

- 88E8053-XX-NNC-C000

Figure 25 shows a typical package marking and pin 1 location for an 88E8053 part. Markings for the other variants are similar.

Figure 25: 88E8053 Package Marking and Pin 1 Location



MARVELL®

MOVING FORWARD FASTER®

Marvell Semiconductor, Inc.

700 First Avenue
Sunnyvale, CA 94089

Phone 408.222.2500
Fax 408.752.9028

www.marvell.com

US and Worldwide Offices

Marvell Semiconductor, Inc.
700 First Avenue
Sunnyvale, CA 94089
Tel: 1.408.222.2500
Fax: 1.408.752.9028

Marvell Asia Pte, Ltd.
151 Lorong Chuan, #02-05
New Tech Park
Singapore 556741
Tel: 65.6756.1600
Fax: 65.6756.7600

Marvell Japan K.K.
Shinjuku Center Bldg. 50F
1-25-1, Nishi-Shinjuku, Shinjuku-ku
Tokyo 163-0650
Tel: 81.(0).3.5324.0355
Fax: 81.(0).3.5324.0354

Marvell Semiconductor Israel, Ltd.
Moshav Manof
D.N. Misgav 20184
Israel
Tel: 972.4.999.9555
Fax: 972.4.999.9574

Worldwide Sales Offices

Western US Sales Office
Marvell
700 First Avenue
Sunnyvale, CA 94089
Tel: 1.408.222.2500
Fax: 1.408.752.9028
Sales Fax: 1.408.752.9029

Central US Sales Office
Marvell
11709 Boulder Lane, Ste. #220
Austin, TX 78726
Tel: 1.512.336.1551
Fax: 1.512.336.1552

Eastern US/Canada Sales Office
Marvell
Parlee Office Park
1 Meeting House Road, Suite 1
Chelmsford, MA 01824
Tel: 978 250-0588
Fax: 978 250-0589

Europe Sales Office
Marvell
3 Clifton Court
Corner Hall
Hemel Hempstead
Hertfordshire, HP3 9XY
United Kingdom
Tel: 44.(0).1442.211668
Fax: 44.(0).1442.211543

Marvell
Fagerstagatan 4
163 08 Spanga
Stockholm, Sweden
Tel: 46.16.146348
Fax: 46.16.482425

Marvell
5 Rue Poincare
56400 Le Bono
France
Tel: 33.297.579697
Fax: 33.297.578933

Israel Sales Office
Marvell
Ofek Center Bldg. 2, Floor 2
Northern Industrial Zone
LOD 71293
Israel
Tel: 972.8.924.7555
Fax: 972.8.924.7554

China Sales Office
Marvell
5J, 1800 Zhong Shan West Road
Shanghai, China 200233
Tel: 86.21.6440.1350
Fax: 86.21.6440.0799

Japan Sales Office
Marvell
Helios Kannai Bldg. 12F
3-21-2 Motohama-cho, Naka-ku
Yokohama, Kanagawa
Japan 231-0004
Tel: 81.45.222.8811
Fax: 81.45.222.8812

Taiwan Sales Office
Marvell
2Fl., No. 1, Alley 20, Lane 407
Ti-Ding Blvd., Nei Hu District
Taipei, Taiwan 114, R. O. C
Tel: (886-2).7720.5700
FAX: (886-2).7720.5707

For more information, visit our website at: www.marvell.com