SoC*e*

System-on-Chip *engineering*

ManagedEthernet*Switch* (**ME***S*)
IP Core Datasheet

- Confidential -

Core:1909 (Rev. 0) Doc: 190911

This page has been intentionally left blank

Copyright reminder

Revision History

Rev.	Date	Author	Description
171121	17/11/21	EO	Initial version of the new IP.
171219	17/12/19	EO	DLR Ring supervisor.
180220	18/02/20	EO	Version 18.02.
180327	18/03/27	EO	Version 18.04.
190911	19/09/11	EO	Version 19.09.

Contents

List of Figures

List of Tables

1. Acronyms

The list below contains abbreviations for terms used throughout this document.

AXI	Advanced eXtensible Interface
COE	Configuration Over Ethernet
CRC	Cyclic Redundancy Check
DLR	Device Level Ring
DSA	Distributed Switch Architecture
DSCP	Differentiated Services Code Point
E2E	End-to-end
FIFO	First Input First Output
FPGA	Field Programmable Gate Array
FSM	Finite State Machine
GMII	Gigabit Media Independent Interface
HSR	High-availability Seamless Redundancy
ICAP	Internal Configuration Access Port
IIC	Inter-Integrated Circuit
MAC	Media Access Control
MDIO	Management Data Input/Output
MES	Management Ethernet Switch
MII	Media Independent Interface
MRC	Media Redundancy Client
MRM	Media Redundancy Manager
MRP	Media Redundancy Protocol
MSB	Most Significant Bit
PCP	Priority Code Point
PHY	Physical Layer
PRP	Parallel Redundancy Protocol
PTP	Precise Time Protocol
P2P	Peer-to-peer
QoS	Quality of Service
RGMII	Reduced Gigabit Media Independent Interface
RSTP	Rapid Spanning Tree Protocol
SFD	Start of Frame Delimiter
TOS	Type of Service
UART	Universal Asynchronous Receiver-Transmitter
VLAN	Virtual Local Area Network

2. Introduction

This document describes **ManagedEthernet***Switch* IP core (**ME***S*) of **SoC***e* (Figure 2.1). This IP is implementable on programmable hardware platforms and requires external configuration. It has been designed to address the maximum throughput using the minimum resources.

The core of the switch is a non-blocking crossbar matrix that allows continuous transfers between all the ports. It implements Store&Forward switching approach in order to fulfil Ethernet standard policy regarding frame integrity checking each frame before forwarding them. However the latency time has been minimized to nanoseconds order. Thus, **ME***S* is the perfect switch to implement in Ethernet based Industrial Networks.

ME*S* can be used in combination with **SoC***e* **HSR-PRP***Switch* IP to introduce HSR and PRP capabilities in the ports that were required. HSR switching approach is Cut-Through. Thus, the combination of **ME***S* and **HSR-PRP***Switch* will offer the maximum performance and maximum compatibility with the standards.

It is supported on Xilinx Spartan-6, 7-Series and UltraScale+ MPSoC devices.

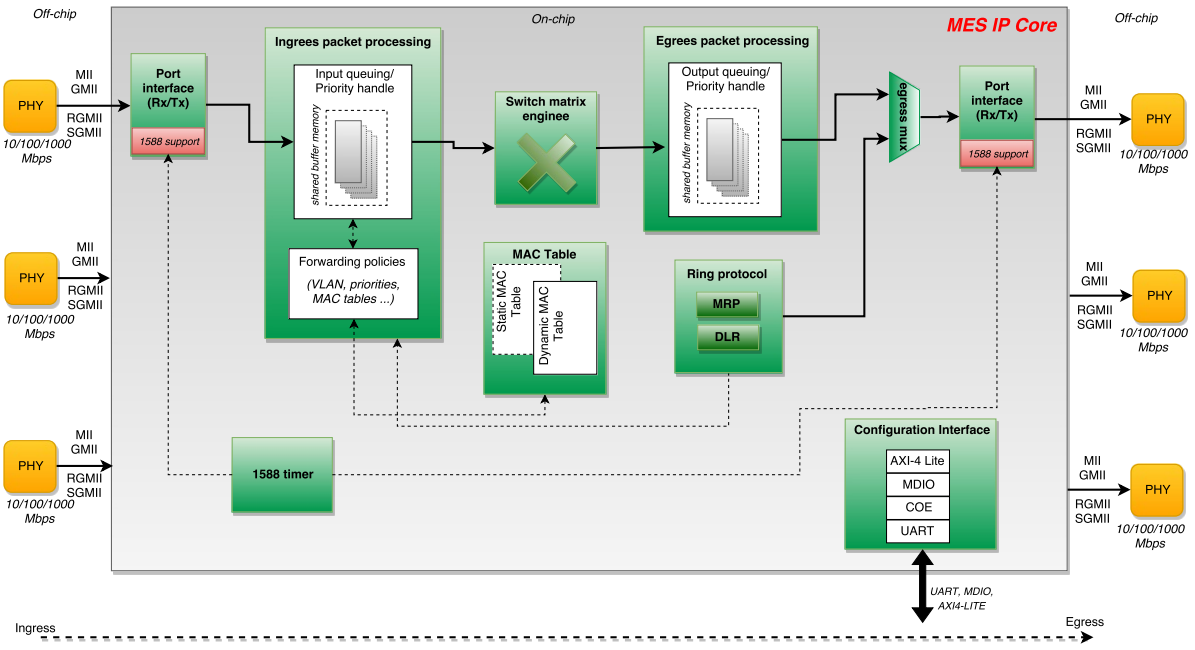


Figure 2.1.: **ManagedEthernet***Switch* **IP core block diagram.**

3. General Features

The standard features that the **MES** IP core includes are:

- Configurable number of Ethernet ports (up to 16 ports and depends on the available resources).
- Full-Duplex (R)GMII/MII and Half-Duplex MII interfaces for attaching to an external Physical Layer device (PHY). Possible to work with different data rate (10/100/1000 Mbps) in each port. It also offers the possibility to include SGMII functionality by internally connecting its GMII side to the Ethernet 1000BASE-X PCS/PMA core from Xilinx.
- Dynamic MAC Table (up to 4096 entries). Automatic MAC addresses learning and ageing.
- Static MAC addresses (up to 4096 entries).
- Mechanism of providing the list of MAC Addresses that are contained in its Dynamic and Static MAC Tables classified by the port number of the switch.
- Port-based VLAN support.
- UART, MDIO, COE (Configuration Over Ethernet) or AXI4-lite Configuration interfaces.
- IEEE 1588 StateLess Transparent Clock functionality (End-to-End or Peer-to-peer) in layer 2.
- Quality of Service according the PCP bits (802.1p), the DSCP TOS bits of the IP packet or Ethertype.
- Jumbo frame support. Note that the presence of jumbo frames may have an adverse effect on network and also in the resource usage in the FPGA.
- Rapid Spanning Tree Protocol (RSTP) support.
- 802.1x support.
- Media Redundancy Protocol (MRP) in client (MRC) or in manager (MRM) mode.
- Per port frame rate limiting.
- Broadcast storm protection.

- Multicast frame filtering.
- Port mirroring.
- Device Level Ring (DLR) in beacon based or ring supervisor mode.
- DSA (Distributed Switch Architecture) for frame tagging.
- Statistics Counters for managing and debugging purposes.

4. Functional Considerations

4.1. PHY interface

4.1.1. RGMII interface

The **ME**S IP core supports RGMII interface natively. However, it is mandatory that the PHY device inserts the necessary delay (1.2 to 2 ns) between the clock and data lines in the transmission path (from FPGA to PHY). In reception path (from PHY to FPGA) it is recommended that the PHY device inserts the necessary delay between the clock and data lines. However, if the PHY is not capable of performing such delay, it is possible to do it by means of *IODELAY* capabilities. Please note that the target FPGA should support *IODELAY* management.

4.1.2. SGMII interface

The **ME**S IP core supports SGMII interface by internally connecting its GMII side to the Ethernet 1000BASE-X PCS/PMA core from Xilinx.

4.2. MAC address learning

4.2.1. Dynamic MAC Table

The *Dynamic MAC Table* learns automatically MAC addresses based on the source address of the received frames. The entries are removed from the MAC address table if the specified ageing time has exceeded. If the source port of an entry changes, this entry is updated.

4.2.2. Static MAC Table

The switch dynamically builds the address table by using the MAC source address of the frames received. However, it is also possible to enter MAC addresses, which are termed static MAC addresses, into the *Static MAC Table*. Contrary to what happens in the *Dynamic MAC Table*, in the *Static MAC Table* is possible to assign more than one port to each static MAC address as an origin port (used to know to which port(s) forwards frames). It is useful when multicast addresses (multicast filtering) have to be configured.

The *Static MAC Table*, when it is enabled, is a table that runs in parallel with the *Dynamic MAC Table*. When a MAC consultation is performed, both dynamic and static MAC tables are accessed at the same time.

4.3. IEEE 1588 v2 Transparent Clock

ME*S* IP core supports IEEE 1588 StateLess TC (Transparent Clock) in E2E (End-to-End) or P2P (Peer-to-Peer) and in layer 2 enabling precise synchronization of device clocks in packet based networks. Basically, it consists in timestamping PTP event frames in the ingress and egress ports and calculating the delay caused by the switch. The link delay calculated by the Peer-to-peer mechanism is also added to the correction field of the PTP event frames.

4.4. Quality of Service (QoS)

ME*S* IP core can handle priorities on all ingress ports. It supports the following Quality of Service schemes. Note that these schemes are exclusive and different schemes can be defined for each port:

- Performs a strict QoS according to the PCP bits (802.1p).
- QoS according to the DSCP TOS bits of the IP packet.
- QoS based on Ethertype.

The number of applicable priority queues is configurable at implementation time. The possible number of queues are 2, 4 or 8. In that sense, it uses logic equivalences in order to accommodate each priority into the corresponding queue as tables 4.1 and 4.2 show. As an example four queues are used to show these equivalences.

Table 4.1.: **PCP bits to queue equivalence.**

PCP value [2..0]	Queue
000	0
001	0
010	1
011	1
100	2
101	2
110	3
111	3

Table 4.2.: **TOS bits to queue equivalence.**

DSCP Name	DSCP value	IP precedence	Queue
CS0	000000	0	0
CS1	001000	1	0
AF11	001010	1	0
AF12	001100	1	0
AF13	001110	1	0
CS2	010000	2	1
AF21	010010	2	1
AF22	010100	2	1
AF23	010110	2	1
CS3	011000	3	1
AF31	011010	3	1
AF32	011100	3	1
AF33	011110	3	1
CS4	100000	4	2
AF41	100010	4	2
AF42	100100	4	2
AF43	100110	4	2
CS5	101000	5	2
EF	101110	5	2
CS6	110000	6	3
CS7	111000	7	3

MES IP core applies priorities at port level among the queues at ingress and egress paths, egressing first those frames stored in the higher priority buffer (strict prioritization).

4.5. VLAN Support - Port-based Virtual LAN

MES IP core supports Port-based Virtual LANs (VLANs), up to 4096 VLANs. In this way, the switch can be virtually divided into virtual switches. Frames from a port cannot be forwarded into a port of another VLAN. So, VLAN configuration affects in the forwarding decision.

You can configure two basic roles for ports on **MES** IP core:

- *Access port (untagged port)*: it belongs only to the configured access VLAN and is designed to provide connectivity for an end device. In access mode, there is only one VLAN configured with the Vlan ID 1 as default.
- *Trunk port (tagged port)*: this port is used to interconnect switches. It allows to devices attached to a particular switch to communicate with devices attached to another switch. The trunk ports can transmit traffic from multiple VLANs configured locally across the trunk (over a single connection), thus allowing a VLAN to be distributed over multiple switches.

The ports perform different functions depending on the configured Port mode:

- Access port:
  1. If frame does not carry VLAN tag, it is tagged with the native VLAN ID.
  2. If frame carries VLAN tag (it should not carry),
     - if it does not match with the port's VLAN ID, it is removed.
     - if it matches with the port's VLAN ID, it is tagged again.
  3. When transmitting, VLAN tag is always removed.
- Trunk port:
  1. If frame does not carry VLAN tag, it is tagged with the native VLAN.
  2. If frame carries VLAN tag, it is not tagged again.
     - if it does not match with the port's allowed VLAN IDs, it is removed.
     - if it matches with the port's allowed VLAN IDs, it is accepted.
  3. When transmitting, the VLAN tag is not removed.

Note: The minimum frame length of a non VLAN tagged frame is 60 bytes (without CRC) whereas a VLAN tagged frame is 64 bytes (without CRC). In both case, if short frames are received, they are padded until these minimum lengths are reached.

4.6. Switching Portmask

Apart from Port-based VLAN, **MES** provides a special forwarding configuration mode using the *Switching mask* register. It allows to distribute frames in the switch more versatilely as it performs the same functionality that Port based VLAN does, and it also allows to force the forwarding of frames to concrete ports.

4.7. Ethertype based switching

This feature enables to transmit specific traffic through specific ports based on its Ethertype. The remaining traffic is switched according to the switching standard policy. This feature can be configured independently for each port.

4.8. RSTP Support

MES IP supports Rapid Spanning Tree Protocol (RSTP). RSTP is a protocol defined by the IEEE 802.1D standard as an evolution of the STP, in order to be used on a switched redundant network. The main features of the RSTP support included on **MES** IP are:

- It meets all the requirements of the IEEE 802.1D standard related to packet switching.
- It is a mixed *hardware* and external processing *software* solution.
- If offers a very low latency times for the Ethernet frames switching time because of the *hardware* packet switching.
- It allows the processing unit to work as a CPU connected to the switch in order to switch its Ethernet frames.

The RSTP support offered by this IP core is implemented by a mixed *hardware* and external *software* solution. The features that are implemented in the *hardware* section are the following ones:

- Frame reception.
- Frame filtering based on MAC Table (using *mac_addres_table* module).
- Frame filtering based on port state.
- Frame filtering based on a filtering database.
- Frame transmission.

In order to apply RSTP functionality, there has to be a software application in which RSTP algorithm is running. This application communicates to the FPGA the state (discarding, learning or forwarding) of the ports via one of the register interfaces. The FPGA handles frames based in the state of the ports. Independently of the state of ports, the FPGA has to pass the BDPU frames to the application.

In order to implement the full RSTP functionality, combining the Software and Hardware sections, there are many implementation options depending on the customer's CPU. **SoC***e* provides the following Reference Designs to implement full RSTP functionalities using **ME***S* IP core:

- Xilinx Zynq external platform running Linux.
- Microblaze soft-processor running Petalinux on an embedded.

4.8.1. RSTP Support Architecture

The block diagram of the RSTP support included in **ME***S* IP core is depicted in Figure 4.1:

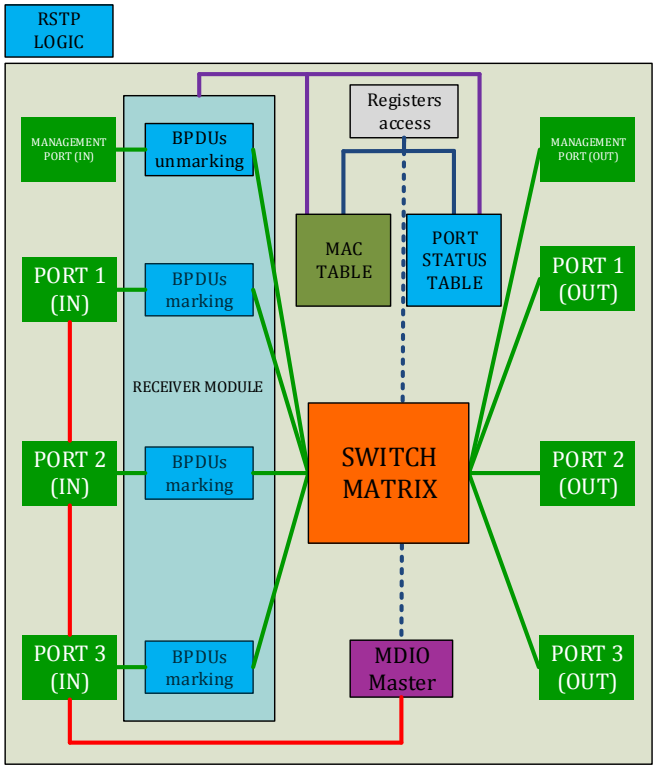


Figure 4.1.: **RSTP support architecture.**

*** BPDU Filter**

This BPDU Filter processes the received BPDUs and sends them to the BPDU marking part.

RSTP requires a management port where a CPU running RSTP stack should be connected. This CPU could be the PS on Zynq devices (optimal solution) but also an external CPU. There are two proposals for the RSTP software:

- GPL SourceForge MSTPd (already included on the basic MES IP purchasing)
- SoC-e POSIX compatible RSTP stack (can be purchased separately)

This is based on the destination MAC address that in the case of BPDUs is: 01:80:C2:00:00:00.

*** Port State Based Filter**

This part of the core performs the Ethernet frame filtering based on the state of the port. This state must be one of the following:

- Discarding state: In this state, all frames, except the BPDUs are discarded.
- Learning state: In this state, all frames except the BPDUs are discarded but its source MAC address is learned.
- Forwarding state: In this state, all frames are forwarded, and also its MAC address is learned.

4.8.2. Filtering Database Filter

This part of the core performs the filter of the frames, based on destination MAC address. Each database entry has a lifetime and after that lifetime it is discarded. Database's entries are set based on the MAC addresses learned on learning and forwarding states.

4.8.3. BPDUs Marking

This part hardware implementation performs the filtering and marking of the BPDUs in order to send them to the external CPU. The BPDUs that are received on one of the two switch ports are not forwarded to the other port. Instead, they are marked, based on what port do they come from. The byte that is used in order to mark the frames is the destination address last byte. Since all BPDUs have same destination address here the following action is performed

- If the BPDU has been received on port 0, its last destination MAC address byte is set to 0x20.
- If the BPDU has been received on port 1, its last destination MAC address byte is set to 0x21.

- If the BPDU has been received on port N, its last destination MAC address byte is set to 0x2N.

Then the frame is forwarded to the external CPU

4.8.4. BPDUs Unmarking

This hardware processing unit is in charge of performing the filtering and unmarking of the BPDUs in order to send them to other switches with RSTP support. The BPDUs that are received from the external cpu port are not forwarded to switch ports. instead of it, the unmarking is done and based on the destination MAC address last byte the frames are forwarded to one of the two switch ports:

- If the BPDU has its last destination MAC address byte set to 0x20, it is set to 0x00 and then the BPDU is sent by port 0
- If the BPDU has its last destination MAC address byte set to 0x21, it is set to 0x00 and then the BPDU is sent by port 1
- If the BPDU has its last destination MAC address byte set to 0x2N, it is set to 0x00 and then the BPDU is sent by port N

4.8.5. Interconnection Interfaces

As can be extracted from the Block Diagram presented in Figure 4.1, there are two interfaces between the IP core and the external customer CPU:

- One Ethernet interface
- One Register access interface

* Ethernet Interface

The Ethernet interface is used in order to send and receive Ethernet frames(including marked BPDUs) between the IP core and the external CPU.

* Register access Interface

The Register access interface is used in order to receive from the external CPU the state of each port. This interface can be UART, MDIO, COE or AXI.

4.8.6. RSTP External CPU Requirements

In order to be able to use correctly this RSTP implementation, the external CPU needs to perform the following actions.

- It must be able to receive and to identify BPDUs from one Ethernet interface.

- It shall distinguish between both interfaces of the switch based on BPDUs destination MAC address.
- It must process the RSTP algorithms based on the received BPDUs.
- It must check the state of the external ports Link by accessing the PHYs by the use of the MDIO master.
- It must generate and mark BPDUs packets them as defined for port identification.
- It must advice the IP core when a port state changes attending to BPDU processing. This operation is done though Register access interface (Table **??**).
- It must advice the IP core when a topology change BPDU has arrived so the IP core has to flush its MAC table. This is done by performing a write to the RSTP control register bit 0.

4.9. IEEE 802.1X Support

ME*S* IP supports IEEE 802.1X. It is an standard for port-based Network Access Control. It provides authentication mechanism to devices wishing to attach to a LAN/WLAN. The protocol in 802.1X is called EAP encapsulation over LANs (EAPOL). The main features of the IEEE 802.1X support included on **ME***S* IP are:

- **Port Blocking**: It blocks the port according to external signals while allowing 802.1X traffic (EAPOL) to pass through the device. One register is used for each port.
- **MAC based Blocking**: It features a MAC filtering option where only selected MACs will pass the core. In any case EAPOL traffic will pass the IP Core.
- **Transparent Mode**: In this mode, all the traffic passes through the device.
- **User configurable list of allowed MAC addresses**: Two registers are used for each mac and port.

4.10. Per Port Frame Rate Limiting

The *Per Port Frame Rate Limiting* functionality provides a mechanism to limit the maximum number of frames in the ingress path. When this feature is enabled, a threshold level (different depending on the PHY speed) is set at which the control is applied. Different threshold levels can be applied for each frame type (broadcast/multicast/unicast) and for each port. Thanks to this functionality if the rate at which frames arrive at a port exceeds a defined limit, the switch will block such packets at that port.

As said before, it is possible to configure the maximum number of frames allowed in the configured time window. The check windows depend on the speed of the PHY and are:

- **In 10 Mbps**: 100 ms
- **In 100 Mbps**: 10 ms
- **In 1000 Mbps**: 1 ms

4.10.1. Broadcast Storm protection

The *Per Port Frame Rate Limiting* functionality is used for *broadcast storm* protection purposes. The *broadcast storm* results in performance degradation and often causes network choking. These storms come from errors or a network loop and may be created by an application on one node. The *Per Port Frame Rate Limiting* functionality gives the possible to control these broadcast storms setting a threshold level at which the switch starts blocking those packets.

4.11. Multicast filtering

This functionality enables to flood only previously configured multicast traffic and to specific ports. The rest of the multicast traffic is filtered out by the switch. In order to use multicast address filtering functionality, it is necessary to use the *Static MAC table*.

The multicast addresses that are needed to be filtered are configured in the *Static MAC table*. Note that, unlike with unicast MAC addresses, more than one source port can be configured for each multicast address.

On the other hand, when a search process of a multicast MAC address in the *Static MAC Table* reports that there is not match, the result is that this multicast frame has to be dropped. This behaviour is different in the case of unicast MAC addresses, because if there is not match, this unicast MAC address is flooded to all ports.

4.12. Port mirroring

This functionality enables to copy packets entering a port or ports ("mirrored port(s)") and to send the copies to a monitor port ("mirror port") for local monitoring or remote monitoring. Port mirroring enables the network administrator to keep close track of switch performance (monitoring compliance, enforcing policies, detecting intrusions, monitoring and predicting traffic patterns, correlating events, and so on) by placing, for example, a protocol analyser on the port that is receiving the mirrored data.

It is important to keep performance in mind when configuring port mirroring. For example, if you mirror traffic from multiple ports, the mirrored traffic may exceed the capacity of the output interface.

4.13. Media Redundancy Protocol (MRP) - MRC/MRM

MES IP supports the Media Redundancy Protocol (MRP), a recovery protocol based on a ring topology, and the redundant interconnection of MRP rings via the Media Redundancy Interconnection Protocol. A MRP compliant network shall have a ring topology with multiple nodes (MRM and MRC).

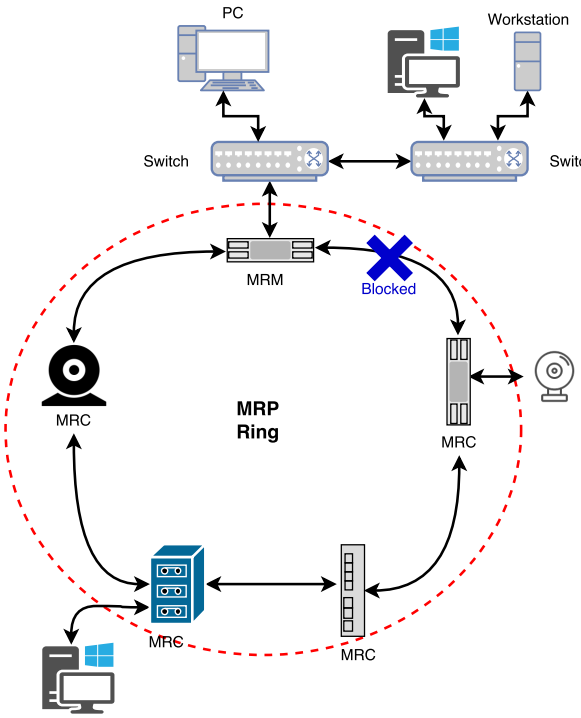


Figure 4.2.: **MRP Network.**

4.13.1. Media Redundancy Manager (MRM)

One of the nodes in a MRP network has the role of a Media Redundancy Manager (MRM). The function of the MRM is to observe and to control the ring topology in order to react on network faults.

The MRM node performs the following functions:

- The MRM shall control the ring state by:

  - Sending *MRP_Test* frames at a configured time period in both directions of the ring.
  - Setting one ring port to FORWARDING state and the other ring port to BLOCKED state if it receives its own MRP_Test frames.
  - Setting both ring ports to FORWARDING state if it does not receive its own *MRP_Test* frames on both ring ports within a configured time according to *MRP_TSTdefaultT*, *MRP_TSTshortT* and *MRP_TSTNRmax*.

- Transmitting *MRP_TopologyChange* frames, the MRM indicates changes in the ring state to the MRCs.
- The MRM does not forward MRP specific frames between its ring ports. Those frames are *MRP_Test* frames, *MRP_TopologyChange* frames, *MRP_LinkChange* frames.

4.13.2. Media Redundancy Client (MRC)

Nodes in a MRP ring that do not act as the ring managers have the role of Media Redundancy Clients (MRC). These nodes react on received reconfiguration frames from the MRM and can detect and signal link changes on its ring ports.

The MRC node performs the following functions:

- It detects a failure or recovery of a ring port link and notifies the change by sending *MRP_LinkChange* frames through both of its ring ports.
- It processes the *MRP_TopologyChange* frames and clears its FDB if requested by an *MRP_TopologyChange* frame in a given time interval MRP_Interval.
- It forwards *MRP_Test* frames, *MRP_TopologyChange* frames, *MRP_LinkChange*, *MRP_InTest* frames, *MRP_InLinkChange* frames, *MRP_InTopologyChange* frames and *MRP_InLinkStatusPoll* frames between its ring ports only.
- The ring ports take one of the following port states:
  - **BLOCKED**: all frames are dropped except the MRP frames.
  - **FORWARDING**: all frames are passed through according to the forwarding rules.

4.14. Device Level Ring (DLR)

ME*S* IP supports the DLR protocol (IEC 61158-4-2:2014, clause 10) which provides high network availability in a ring topology and provides fast network fault detection and reconfiguration. Since the DLR protocol operates at Layer 2, the presence of the ring topology and the operation of the DLR protocol are transparent to higher layer protocols.

A DLR network includes at least one node configured to be a ring supervisor, and any number of Beacon Based nodes (Figure 4.3). It is assumed that all the ring nodes have at least two Ethernet ports and incorporate embedded switch technology.

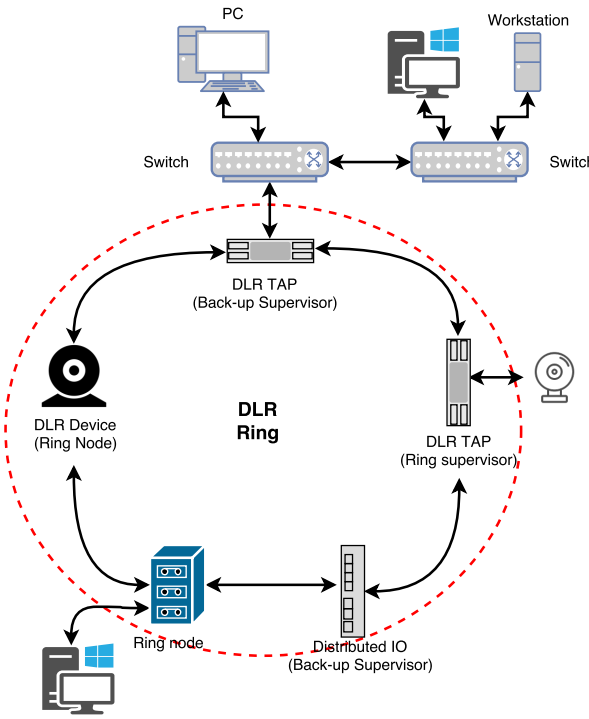


Figure 4.3.: **DLR Network.**

4.14.1. Ring Supervisor node

A DLR network requires that at least one device be configured to act as the Ring Supervisor. This class of device shall implement the required ring supervisor behaviours such as verifying the integrity of the ring, reconfiguring the ring to recover from faults, collecting diagnostic information for the ring or sending and processing Beacon frames at the default beacon interval. The active Ring Supervisor blocks traffic on one of its ports (with the exception of a few special frames) and does not forward traffic from one port to another, thus avoiding a network loop.

4.14.2. Beacon-based ring node

A ring node or Beacon-based node is any non-supervisor device that operates on the ring and participates in the DLR protocol (faults detection), but without the ring supervisor capability. These beacon-based ring nodes are required to process beacon frames sent by a ring supervisor within a specified beacon interval.

4.15. Distributed Switch Architecture (DSA)

This hardware feature is used to insert a specific tag for each Ethernet frames it received to/from specific ports to help the management port figure out:

- What port is this frame coming from.
- How to send CPU originated traffic to specific ports.

4.16. Statistics Counters

During operation, the core collects statistics on the success and failure of various operations for processing by network management entities elsewhere in the system. These statistics are accessed through the different management interfaces offered by the IP core. Statistics counters are described in Tables 8.104 and 8.105. These statistic counters are wraparound counters and have a reset function. They can return to zero when they naturally wrap around, they are reseted or when the device is reconfigured. All statistics counters are read-only, so write attempts to statistics counters will be unsuccessful.

4.17. IP core status interruptions

The **ME***S* IP core provides some bit vectors which give information about the status of the IP core. Each of these bits represent an event in the IP, such as CRC errors, buffer overflows, etc. When any of these bits is asserted and associated interrupt mask bit is set, the *status_interrupt* output will be asserted (active high) as an interrupt event notification.

5. Port Description

5.1. User Interfaces

5.1.1. Clock, Reset and Link Signal Definition

Table 5.1 describes the reset signal, the clock signals and the link signals that are input to the core. The clock signals should be generated in the top-level wrapper of the core.

There is also an general input port called *IP_enable* which is used to enable/disable the IP core.

Table 5.1.: **Clock, Reset and Link Signal Definition.**

Signal	Direction	Clock Domain	Description
clk_in	in	-	Global clock of 156.25 MHz, 125 MHz or 100 MHz frequency (its frequency has to be the same as it is specified in the *System Frequency* customization parameter)
reset (*)	in	N/A	Active-High asynchronous reset.
port_?_link	in	N/A	Link of the ports (active high).
IP_enable	in	N/A	IP enable/disable signal.
secure_chip	inout	N/A	Secure chip signal.
secure_chip_valid	out	N/A	Secure chip check valid flag.
status_interrupt (**)	out	clk_in	Interrupt flag to indicate an event in the status bits of the IP core.
NOTE 1 (*): *This port can not be used as a global reset and only has effect in a few parts of logic of the IP core.* **NOTE 2:** *When Link signals are not available, be sure to connect them to constant value of 1 in order to indicate that there is link in the ports.* **NOTE 3 (**):** *This port is only available when "IP Status Information" is selected in the GUI of the IP core.*			

5.2. Physical Interface Signals

5.2.1. PHY Interface Signal Definition

Tables 5.2 to 5.5 describe the possible interface standards supported, RGMII, GMII and MII, which are typically attached to a PHY module, either off-chip or internally integrated. Additionally, it offers the possibility to include SGMII functionality by internally connecting its GMII side to the Ethernet 1000BASE-X PCS/PMA core from Xilinx.

Table 5.2.: **Optional GMII Interface Signal Pinout.**

Signal	Direction	Clock Domain	Description
port_?_phy_rst_n	out	N/A	PHY reset (active low)
port_?_gmii_txd[7:0]	out	tx_clk	Transmit data to PHY
port_?_gmii_tx_en	out	tx_clk	Data Enable control signal to PHY
port_?_gmii_tx_clk	out	-	Clock to PHY (used for 1000)
port_?_mii_tx_clk	in	-	Clock from PHY (used for 10/100)
port_?_gmii_rx_col	in	N/A	Control signal from PHY
port_?_gmii_rx_crs	in	N/A	Control signal from PHY
port_?_gmii_rxd[7:0]	in	port_?_gmii_rx_clk	Received data from PHY
port_?_gmii_rx_dv	in	port_?_gmii_rx_clk	Data Valid control signal from PHY
port_?_gmii_rx_clk	in	-	Clock from PHY
port_?_gmii_rx_er	in	port_?_gmii_rx_clk	Error control signal from PHY
tx_clk: port_?_gmii_tx_clk used for 1000 and port_?_mii_tx_clk used for 10/100			

Table 5.3.: **Optional MII Interface Signal Pinout.**

Signal	Direction	Clock Domain	Description
port_?_phy_rst_n	out	N/A	PHY reset (active low)
port_?_mii_txd[3:0]	out	port_?_mii_tx_clk	Transmit data to PHY
port_?_mii_tx_en	out	port_?_mii_tx_clk	Data Enable control signal to PHY
port_?_mii_tx_clk	in	-	Clock from PHY
port_?_mii_rx_col	in	N/A	Control signal from PHY
port_?_mii_rx_crs	in	N/A	Control signal from PHY
port_?_mii_rxd[3:0]	in	port_?_mii_rx_clk	Received data from PHY
port_?_mii_rx_dv	in	port_?_mii_rx_clk	Data Valid control signal from PHY
port_?_mii_rx_clk	in	-	Clock from PHY
port_?_mii_rx_er	in	port_?_mii_rx_clk	Error control signal from PHY

Table 5.4.: **Optional RGMII Interface Signal Pinout.**

Signal	Direction	Clock Domain	Description
port_?_phy_rst_n	out	N/A	PHY reset (active low)
port_?_rgmii_txd[3:0]	out	port_?_rgmii_txc	Transmit data to PHY
port_?_rgmii_tx_ctl	out	port_?_rgmii_txc	Control signal to PHY
port_?_rgmii_txc	out	-	Clock to PHY
port_?_rgmii_rxd[3:0]	in	port_?_rgmii_rxc	Received data from PHY
port_?_rgmii_rx_ctl	in	port_?_rgmii_rxc	Control signal from PHY
port_?_rgmii_rxc	in	-	Clock from PHY

Table 5.5.: **Optional XILINX GMII-SGMII Interface Signal Pinout.**

Signal	Direction	Clock Domain	Description
port_?_userclk	in	-	Clock from Xilinx GMII-SGMII core. Its frequency is of 125 MHz.
port_?_sgmii_clk_en	in	port_?_userclk	It is a clock enable. The frequency of the port_?_sgmii_clk_en clock enable signal ensures the correct data rate and correct data sampling between the two devices.
port_?_status_vector	in	-	Status signals of the Xilinx GMII-SGMII core. The *Link Status* (bit 0) and *Speed* (bits 11:10) status signals are used in the IP core.
port_?_speed_is_100	out	-	Speed control for controlling operating speed of SGMII interface.
port_?_speed_is_10_100	out	-	Speed control for controlling operating speed of SGMII interface.
port_?_gmii_txd[7:0]	out	port_?_userclk	Transmit data to PHY
port_?_gmii_tx_en	out	port_?_userclk	Data Enable control signal to PHY
port_?_gmii_rxd[7:0]	in	port_?_userclk	Received data from PHY
port_?_gmii_rx_dv	in	port_?_userclk	Data Valid control signal from PHY
port_?_gmii_rx_er	in	port_?_userclk	Error control signal from PHY
- 1Gbps: *port_?_speed_is_10_100 = '0' and port_?_speed_is_100 = '0'*			
- 100Mbps: *port_?_speed_is_10_100 = '1' and port_?_speed_is_100 = '1'*			
- 10Mbps: *port_?_speed_is_10_100 = '1' and port_?_speed_is_100 = '0'*			

5.2.2. AXI4-Stream Signal Definition

The AXI4-Stream master and slave interfaces (Table 5.6) can be used to rapidly connect AXI4-Stream master/slave IP systems.

Table 5.6.: **Optional AXI4-Stream Interface Signal Pinout.**

Signal	Direction	Clock Domain	Description
port_?_s_axis_data_tvalid	in	clk_in	TVALID indicates that the master is driving a valid transfer. A transfer takes place when both TVALID and TREADY are asserted.
port_?_s_axis_data_tlast	in	clk_in	TLAST indicates the boundary of a packet
port_?_s_axis_data_tdata[7:0]	in	clk_in	TDATA is the primary payload that is used to provide the data that is passing across the interface.
port_?_s_axis_data_tready	out	clk_in	TREADY indicates that the slave can accept a transfer in the current cycle.
port_?_m_axis_data_tready	in	clk_in	TREADY indicates that the slave can accept a transfer in the current cycle.
port_?_m_axis_data_tvalid	out	clk_in	TVALID indicates that the master is driving a valid transfer. A transfer takes place when both TVALID and TREADY are asserted.
port_?_m_axis_data_tlast	out	clk_in	TLAST indicates the boundary of a packet.
port_?_m_axis_data_tdata[7:0]	out	clk_in	TDATA is the primary payload that is used to provide the data that is passing across the interface.

5.2.3. MDIO Signal Definition

Table 5.7 describes the MDIO (MII Management) interface signals of the core for each port when the **Add IO Buffers for MDIO Interface Ports** option is not selected. These signals are typically connected to the MDIO port of an on-chip PHY device.

Table 5.7.: **Optional MDIO Interface Signal Pinout.**

Signal	Direction	Description
port_?_mdc	out	MDIO Management Clock
port_?_mdio_i	in	In data signal for communication with PHY configuration and status.
port_?_mdio_o	out	Output data signal for communication with PHY configuration and status.
port_?_mdio_t	out	3-state control for MDIO signals. '1' signals that the value on MDIO_OUT should be asserted onto the MDIO bus.

Table 5.8 shows the MDIO signals when the **Add IO Buffers for MDIO Interface Ports** option is selected. These signals are typically connected to the MDIO bus which connects to an off-chip PHY device.

Table 5.8.: **Optional MDIO Interface Signal Pinout with I/O Buffers.**

Signal	Direction	Description
port_?_mdc	out	MDIO Management Clock
port_?_mdio	in/out	MDIO bus from the bidirectional I/O buffer inserted by the core. The core internally maps the mdio_i, mdio_o and mdio_t signals to this bidirectional I/O buffer.

When MDIO is selected as the interface to access status and configuration registers or the *PHY Emulation* parameter is enabled, another MDIO interface is created. Table 5.9 shows the MDIO (MII Management) interface signals of the core when the **Add IO Buffers for MDIO Configuration Interface** option is not selected.

Table 5.9.: **Optional MDIO Configuration Interface Signal Pinout.**

Signal	Direction	Description
mdc	out	MDIO Management Clock
mdio_i	in	In data signal for communication with MES IP core configuration and status.
mdio_o	out	Output data signal for communication with MES IP core configuration and status.
mdio_t	out	3-state control for MDIO signals. '1' signals that the value on MDIO_OUT should be asserted onto the MDIO bus.

Table 5.10 shows the MDIO signals when the **Add IO Buffers for MDIO Configuration Interface** option is selected.

Table 5.10.: **Optional MDIO Configuration Interface Signal Pinout with I/O Buffers.**

Signal	Direction	Description
mdc	out	MDIO Management Clock
mdio	in/out	MDIO bus from the bidirectional I/O buffer inserted by the core. The core internally maps the mdio_i, mdio_o and mdio_t signals to this bidirectional I/O buffer.

5.2.4. UART Signal Definition

Table 5.11 describes the UART serial interface signals. If UART interface option in the *Configuration Interface Type* is not selected then the generated core does not have the UART logic nor its input/output ports.

Table 5.11.: **Optional UART Interface Signal Pinout.**

Signal	Direction	Description
uart_rx	in	Reception data.
uart_tx	out	Transmission data.

5.2.5. I2C Signal Definition

Table 5.12 describes the I2C (Inter-Integrated Circuit) interface signals of the core when the **Add IO Buffers for I2C Interface Ports** option is not selected. These signals are typically connected to the I2C port of an on-chip device.

Table 5.12.: **Optional I2C Interface Signal Pinout.**

Signal	Direction	Description
SCL_?	out	I2C Clock.
SDA_?_I	in	I2C input data.
SDA_?_O	out	I2C output data.
SDA_?_T	out	3-state control for I2C signals. '1' signals that the value on I2C_OUT should be asserted onto the I2C bus.
NOTE: *There are up to four I2C interfaces available.*		

Table 5.13 shows the I2C signals when the **Add IO Buffers for I2C Interface Ports** option is selected. These signals are typically connected to the I2C bus which connects to an off-chip device.

Table 5.13.: **Optional I2C Interface Signal Pinout with I/O Buffers.**

Signal	Direction	Description
SCL_?	out	I2C Clock.
SDA_?_IO	in/out	I2C bus from the bidirectional I/O buffer inserted by the core. The core internally maps the SDA_i, SDA_o and SDA_t signals to this bidirectional I/O buffer.
NOTE: *There are up to four I2C interfaces available.*		

5.2.6. AXI4-Lite Signal Definition

Table 5.14 describes the AXI-4 Lite interface used to access the status and configuration registers of the **MES** IP core. If AXI interface option in the *Configuration Interface Type* is not selected then the generated core does not have the AXI logic nor its input/output ports.

Table 5.14.: **Optional AXI4-Lite Signal Pinout.**

Signal	Direction	Clock Domain	Description
s_axi_aresetn	in	s_axi_aclk	Local reset for the clock domain
s_axi_awaddr[31:0]	in	s_axi_aclk	Write Address
s_axi_awvalid	in	s_axi_aclk	Write Address Valid
s_axi_wdata[31:0]	in	s_axi_aclk	Write Data
s_axi_wstrb[3:0]	in	s_axi_aclk	Input Write Strobes
s_axi_wvalid	in	s_axi_aclk	Write Data Valid
s_axi_bready	in	s_axi_aclk	Write Response Ready
s_axi_araddr[31:0]	in	s_axi_aclk	Read Address
s_axi_arvalid	in	s_axi_aclk	Read Address Valid
s_axi_rready	in	s_axi_aclk	Read Data/Response Ready
s_axi_arready	out	s_axi_aclk	Read Address Ready
s_axi_rdata[31:0]	out	s_axi_aclk	Read Data
s_axi_rresp[1:0]	out	s_axi_aclk	Read Response
s_axi_rvalid	out	s_axi_aclk	Read Data/Response Valid
s_axi_wready	out	s_axi_aclk	Write Data Ready
s_axi_bresp[1:0]	out	s_axi_aclk	Write Response
s_axi_bvalid	out	s_axi_aclk	Write Response Valid
s_axi_awready	out	s_axi_aclk	Write Address Ready

6. Customization Parameters

This section includes information about customize parameters of the **ME***S* IP core in the Vivado Design Suite.

Note: Figures in this chapter are illustrations of the Vivado IDE. This layout might vary from the current version.

Note: Ports are numbered from 0 to 15.

The customization parameters are divided in six main sections:

- **General Options:** Main configuration parameters of the IP.
- **Configuration Interface:** Configuration parameters related with the interface used (if any) to configure the IP externally.
- **Additional Options:** Configuration parameters related with the features support by the **ME***S* IP core.
- **Additional Protocols:** Configuration parameters used to enable supported protocols.
- **Advanced Settings:** Advanced configuration parameters.
- **Ports 0-7 and Ports 8-15:** Configuration parameters used to specify the functionality of each port.

6.1. General Options

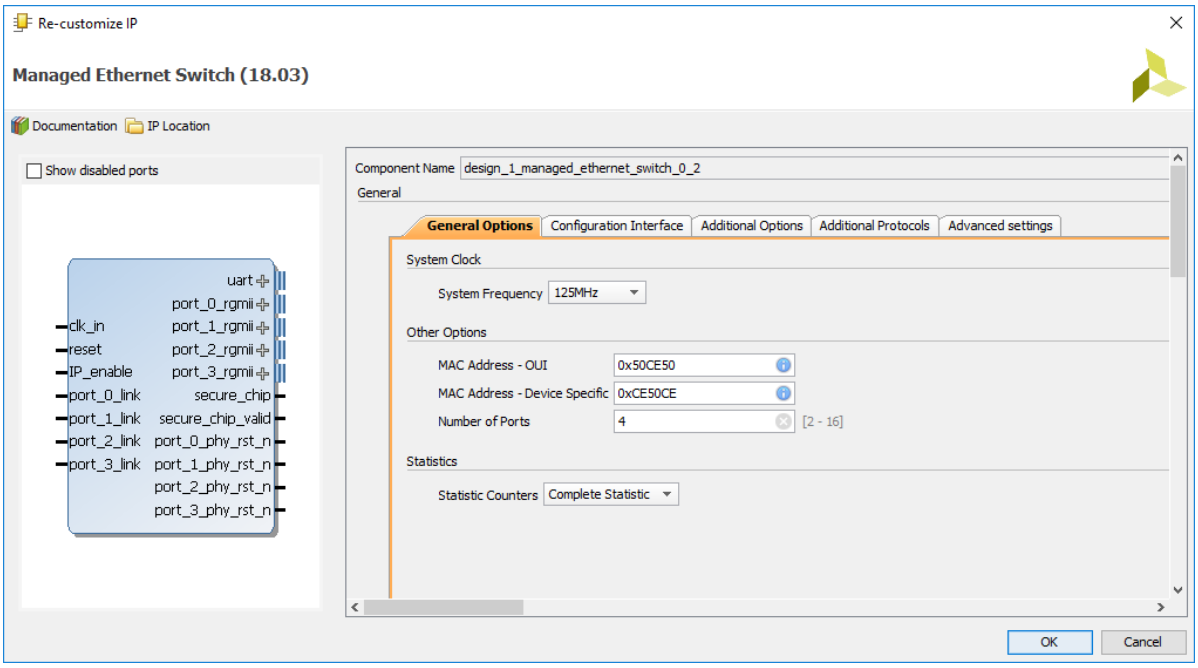


Figure 6.1.: **Vivado Customize IP IDE - General Options Tab**

6.1.1. System Clock

- **System Frequency:** It specifies the system clock frequency used in the IP core. It is selectable between 100 MHz, 125 MHz and 156.25 MHz. Note that if 1000 Mbps Ethernet interface is used, the 125 MHz or 156.25 MHz frequency has to be selected.

6.1.2. Other Options

- **MAC Address - OUI** - It is the first three octets of a default MAC address which uniquely identifies a vendor or manufacturer (OUI - Organizationally Unique Identifier).
- **MAC Address - Device Specific** - It is the last three octets of a default MAC address. The complete MAC address is composed of *MAC Address - Device Specific* and *MAC Address - OUI* fields and is used by the IP core as the source address for any outbound frames generated by the IP itself.
- **Number of Ports** - This specifies the total number of ports of the **ME**S IP core.

6.1.3. Statistics

- **Statistic Counters:** The **MES** IP core can be implemented with optional Statistics Counters. These Statistic Counters can be basic (*Basic Statistic Counters*) or more complete (*Complete Statistic Counters*) for more advanced diagnosis purposes. Note that it has a great impact on the resource usage in the IP core.

6.2. Configuration Interface

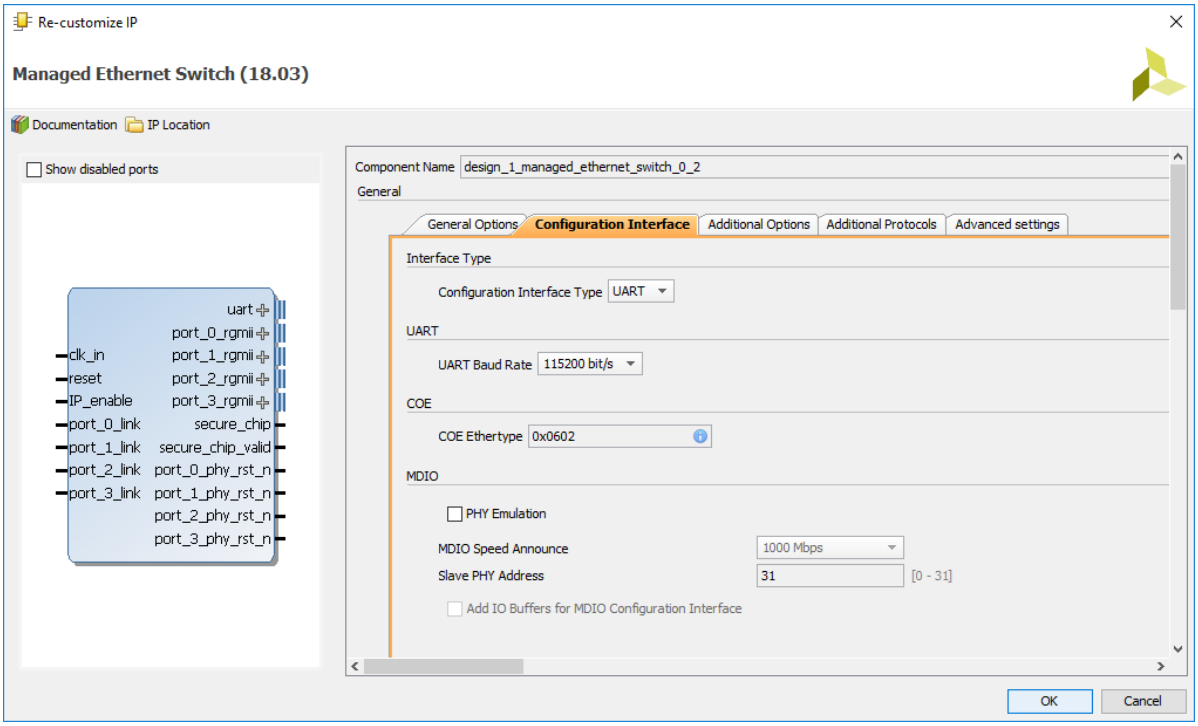


Figure 6.2.: **Vivado Customize IP IDE - Configuration Interface Tab**

6.2.1. Interface Type

- **Configuration Interface Type:** It specifies the interface type used to access configuration and status registers.

6.2.2. UART

- **UART Baud Rate:** It allows the user to select the desired speed for the UART from a list of available baud rates.

6.2.3. COE

- **Ethertype:** When COE interface is selected under the *Configuration Interface Type* parameter, this parameter specifies the Ethertype used for the COE frames. By default, it is set to 0x"0602".

6.2.4. MDIO

- **PHY Emulation:** It specifies if an internal MDIO slave interface is implemented to provide a "fake mii table" at *Slave PHY Address* that allows direct MAC to MAC connection without PHYs to an external CPU. When MDIO interface is selected in the *Configuration Interface Type*, this module is also automatically implemented.
- **MDIO Speed Announce:** When MDIO interface is selected in the *Configuration Interface Type* or *PHY Emulation* is enabled, this parameter specifies the speed that the MDIO module will announce.
- **Slave PHY Address:** When MDIO interface is selected in the *Configuration Interface Type*, this parameter specifies the PHY address in which a "fake mii table" will response.
- **Add IO Buffers for MDIO Interface Port** - If MDIO is enabled as configuration interface, select the Add IO Buffers for MDIO Interface Port option to insert I/O buffer for the MDIO interface port. This creates the bidirectional I/O bus mdio and inserts an output buffer for mdc. If this option is not selected then the core is generated with *mdio_i*, *mdio_o* and *mdio_t* ports.

6.3. Additional Options

6.3.1. Priorities

- **Priority Enable** - **ME***S* IP core is able to handle priorities according to the selected Quality of Service scheme.
- **Priority Levels** - When priorities are enabled, this specifies the number of priority levels to handle.
- **Priority by Ethertype** - When priorities are enabled, this specifies if priorities are based on the Ethertype.

6.3.2. Additional Features

- **Jumbo Frame Enable** - **ME***S* IP core accepts frames over the specified IEEE 802.3-2008 maximum legal length. When it is not enables, the IP only accepts

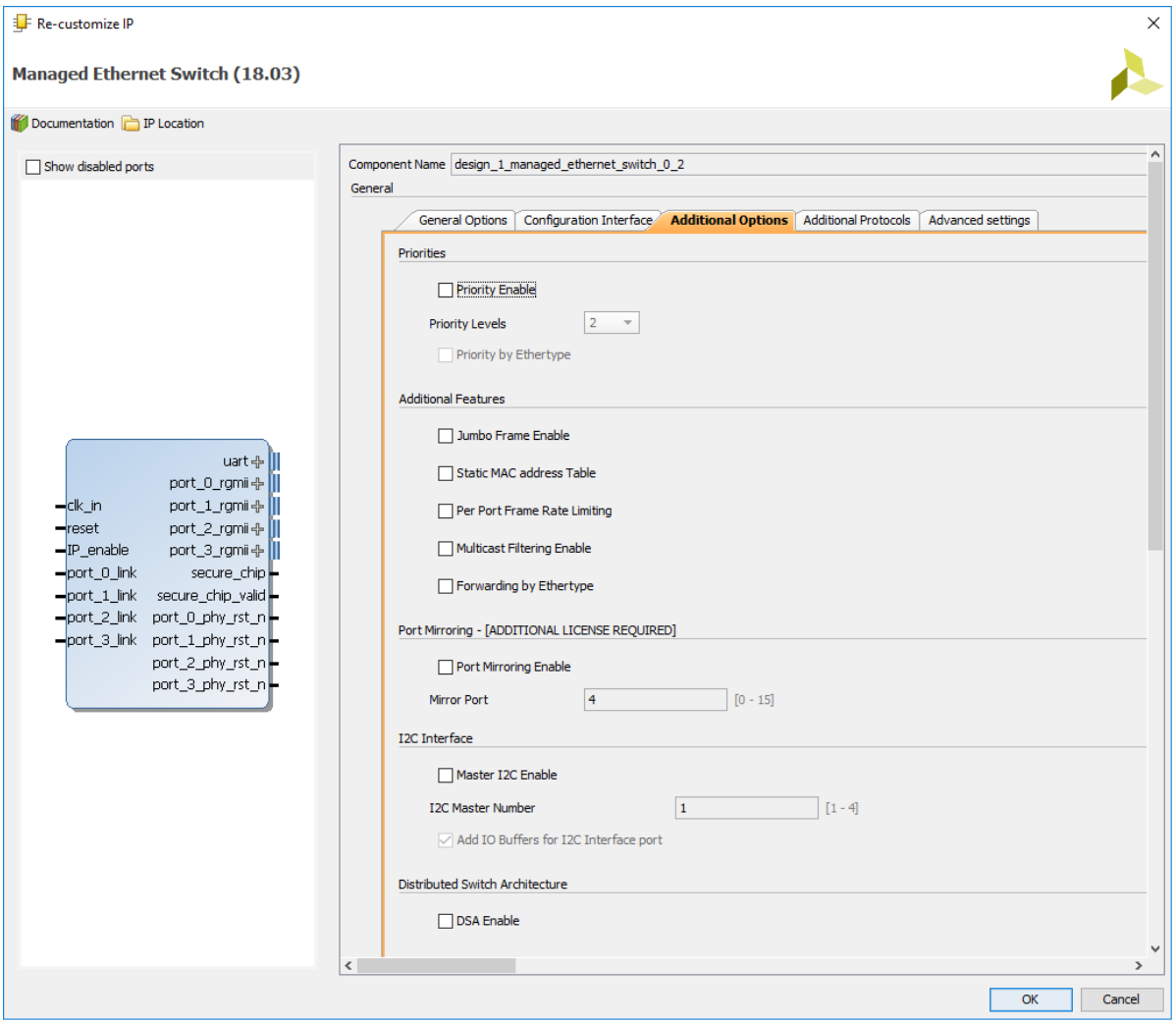


Figure 6.3.: **Vivado Customize IP IDE - Additional Options Tab**

frames up to the specified maximum length (1518 bytes - without preamble and CRC fields).

- **Static MAC Address Table** - **MES** IP core enables to enter static MAC addresses.
- **Per Port Rate Limiting** - **MES** IP core controls the broadcast/multicast/unicast frames arrive rate blocking such packets if defined limit is exceeded.
- **Multicast Filtering Enable** - **MES** IP core filters out those multicast frames that are not configured.
- **Forwarding by Ethertype** - **MES** IP core forwards frames based on the Ethertype.

6.3.3. Port Mirroring

- **Port mirroring enable** - **MES** IP core enables to copy packets entering a port or ports ("mirrored port(s)) and to send the copies to a monitor port ("mirror port") for local monitoring or remote monitoring.
- **Mirror Port** - When *Port mirroring enable* is selected, it defines which port acts as monitor port receiving the copies of packets.

6.3.4. I2C Interface

- **Master I2C Enable** - It enables the I2C Master module.
- **I2C Master Number** - This specifies the number of I2C master modules. Up to four I2C Master modules can be selected.
- **Add IO Buffers for I2C Interface Port** - If I2C is enabled, select the Add IO Buffers for I2C Interface Port option to insert I/O buffers for the I2C interface port. This creates the bidirectional I/O bus sda and inserts an output buffer for scl. If this option is not selected then the core is generated with *sda_i*, *sda_o* and *sda_t* ports.

6.3.5. Distributed Switch Architecture

- **DSA enable** - It enables the Distributed Switch Architecture (DSA) support.

6.4. Additional Protocols

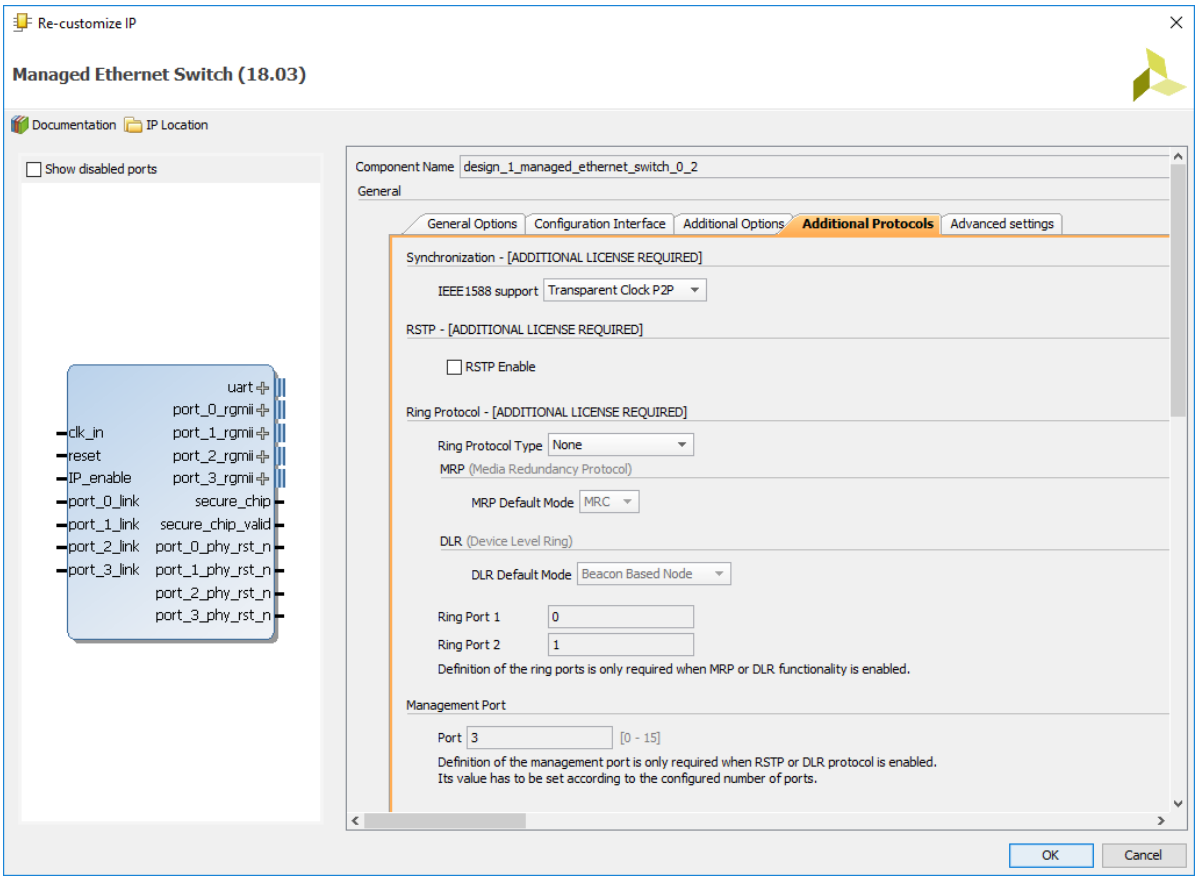


Figure 6.4.: **Vivado Customize IP IDE - Additional Protocols Tab**

6.4.1. Synchronization

- **IEEE1588 Support** - Specify the IEEE1588 synchronization mechanism.
  - **0**: none
  - **1**: Transparent clock End-to-End
  - **2**: Transparent clock Peer-to-Peer

6.4.2. RSTP

- **RSTP Enable** - The core is generated with RSTP (Rapid Spanning Tree Protocol) support which is used to create and maintain the network topology.

6.4.3. Ring Protocol

- **Ring Protocol Type** - The core is generated without any ring protocol, with Media Redundancy Protocol (MRP) support or with Device Level Ring (DLR) support.

  Note: MRP and DLR protocols can not be enabled at the same type.

- **MRP Default Mode** - When *MRP* is selected, it specifies the default MRP mode: MRP Client (MRC) or MRP Manager (MRM).
- **DLR Default Mode** - When *DLR* is selected, it specifies the default DLR node type: Beacon-based node or Ring supervisor.
- **Ring Port 1** - When *MRP* or *DLR* is selected, it defines which port acts as ring port 1.
- **Ring Port 2** - When *MRP* or *DLR* is selected, it defines which port acts as ring port 2.

6.4.4. Management Port

- **Port** - It defines which port acts as management port in those protocols (RSTP and DLR) in which it is necessary. Its value has to be among the number of ports set.

6.5. Advanced Settings

6.5.1. Ultrascale FPGA

- **Family:** When implementing for an Ultrascale FPGA, select one of the available architectures. **Note:** This parameter is not available in the 7 Series version of the IP core.

6.5.2. Frame Storage Queues

- **(*) Ingress Queue Length** - It defines ingress queue size.
- **(*) Egress Queue Length** - It defines egress queue size.

  (*) Notes:

  1. Both the ingress and egress queues use one memory which is shared for all priority queues.
  2. Low queue lengths could have negative effects in the performance.
  3. If Jumbo frame support is enabled, the minimum queue length has to be selected according to the maximum jumbo frame frame.

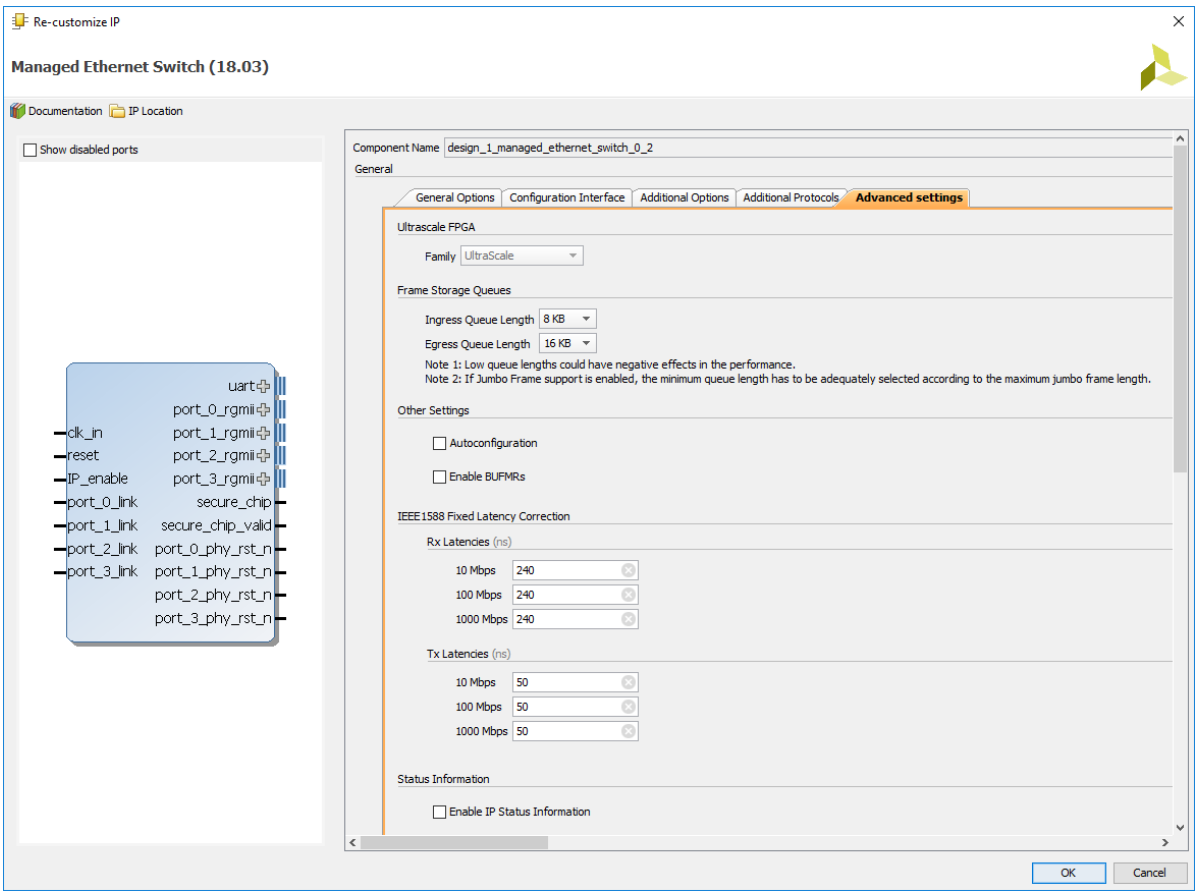


Figure 6.5.: **Vivado Customize IP IDE - Advanced Settings Tab**

6.5.3. Other Settings

- **Autoconfiguration:** It specifies if autoconfiguration is implemented or not. This feature allows **MES** IP core to configure its registers with pre-configured data in the start-up.
- **Enable BUFMRs:** For fully flexible I/O placement, the BUFRs can be replaced by BUFMRs.

6.5.4. IEEE1588 Fixed Latency Correction

Rx Latencies (ns)

- **10 Mbps:** It specifies the default *ns* value relating to the reception latency (10Mbps speed) which is used in the residence time calculation.
- **100 Mbps:** It specifies the default *ns* value relating to the reception latency (100Mbps speed) which is used in the residence time calculation.

- **1000 Mbps:** It specifies the default *ns* value relating to the reception latency (1000Mbps speed) which is used in the residence time calculation.

Tx Latencies (ns)

- **10 Mbps:** It specifies the default *ns* value relating to the transmission latency (10Mbps speed) which is used in the residence time calculation.
- **100 Mbps:** It specifies the default *ns* value relating to the transmission latency (100Mbps speed) which is used in the residence time calculation.
- **1000 Mbps:** It specifies the default *ns* value relating to the transmission latency (1000Mbps speed) which is used in the residence time calculation.

6.5.5. Status Information

- **Enable IP Status Information:** It specifies if the **ME***S* IP core gives information about its status due to events happened during operation.

6.6. Ports 0-7 and Ports 8-15

For each port specified in the **ME***S* IP core, there are the same customization parameters.

- **Interface Type** - Different physical interface types are available for the core:
  - **GMII**: The GMII is defined by the IEEE802.3 specification; it can provide support for Ethernet operation at 10 Mb/s, 100 Mb/s, and 1 Gb/s speeds.
  - **MII**: The MII is defined by the IEEE802.3 specification; it can provide support for Ethernet operation at 10 Mb/s and 100 Mb/s speeds.
  - **RGMII**: The RGMII is, effectively, a Double Data Rate version of GMII; it can provide support for Ethernet operation at 10 Mb/s, 100 Mb/s, and 1 Gb/s speeds.
  - **XILINX GMII-SGMII**: It offers the possibility to include SGMII functionality by internally connecting its PHY side GMII to the Ethernet 1000BASE-X PCS/PMA core from Xilinx.
  - **AXI-STREAM**: AXI4-Stream master and slave interface.
- **Internal Interface** - The core is generated with no physical interface ready for connection to an internal PHY. This feature is only available for *GMII* or *MII* interfaces.
- **Add IOdelay** - This specifies if an internal delay is inserted to the reception part of the selected Ethernet interface. If *Internal Interface* is selected, this option is not available.

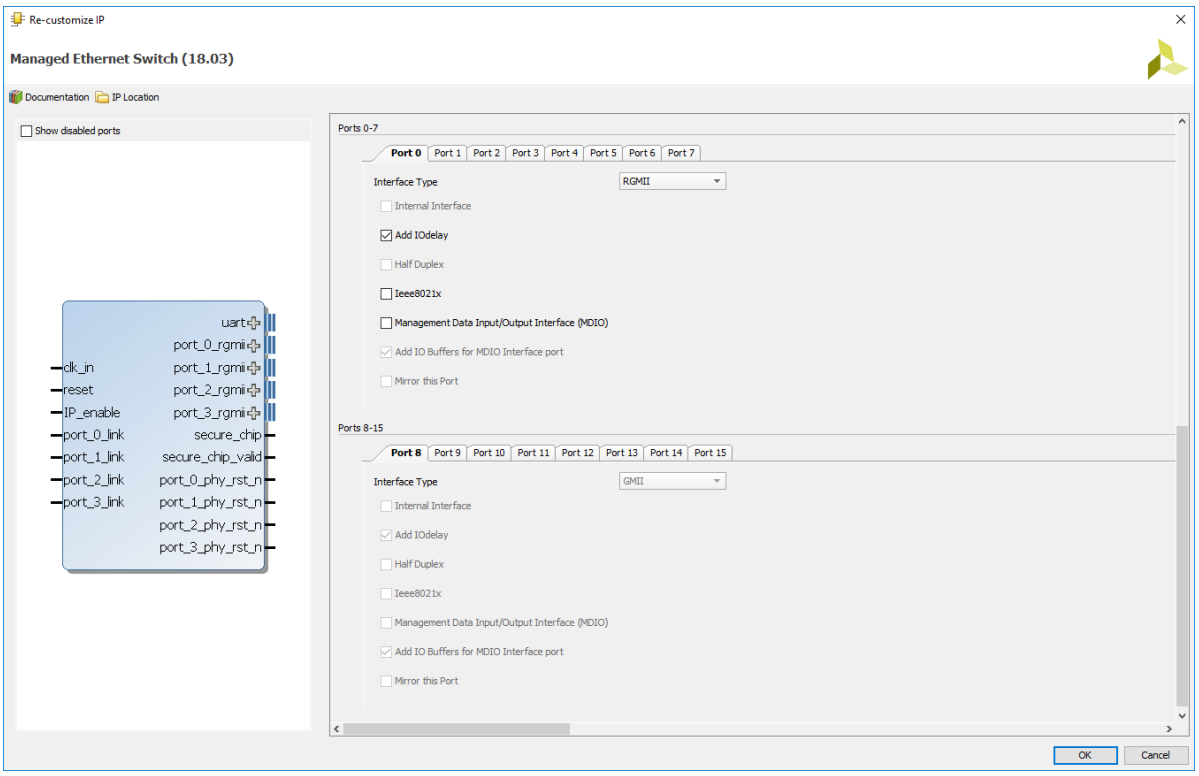


Figure 6.6.: **Vivado Customize IP IDE - Ports 0-7 and Ports 8-15 Tab**

- **Half Duplex** - The **ME***S* IP core always provides support for full-duplex Ethernet (half-duplex is optional). However, to provide half-duplex operation, further FPGA logic resources are required. When the core is generated with half-duplex logic, full- or half-duplex operation can be selected by configuration register. The default is not to include half-duplex support.
- **IEEE8021x** - This specifies if the IEEE802.1X protocol is supported in a port.
- **Management Data Input/Output Interface (MDIO)** - The port has the signals to be connected to the MDIO port of an on-chip or off-chip PHY device.
- **Add IO Buffers for MDIO Interface Port** - If MDIO is enabled, select the Add IO Buffers for MDIO Interface Ports option to insert I/O buffers for the MDIO interface ports. This creates the bidirectional I/O bus mdio and inserts an output buffer for mdc. If this option is not selected then the core is generated with *mdio_i*, *mdio_o* and *mdio_t* ports.
- **Mirror this Port** - When *Port mirroring* is enabled, it specifies if the port will be mirrored. This option is not available for the port set as monitor port (*Mirror Port*).

7. Access to Status and Configuration Registers

Configuration of the core, access to the statistic blocks and access to the MDIO ports, can be provided through different management interface, such as an **UART** interface, a **MDIO** interface or a 32-bit **AXI4-Lite** interface which are independent of the Ethernet datapath. There is also available a management interface called **COE** (Configuration Over Ethernet) which makes use of the Ethernet datapath. The information and configuration of the IP core is the same for any of the interfaces available.

7.1. UART interface

In order to access to the registers via UART, a very simple command interface is provided by the IP core. Table 7.1 summarizes the commands that **ME**S IP core interprets through the UART interface. All commands make use of the ASCII character-encoding scheme and the access can be automated using a simple program running in the CPU or using a standard Terminal Program.

Table 7.1.: **UART commands**

UART Commands		
Type	**Command**	**Description**
Data Read	Rxxxx	Performs a read operation in address $xxxx
	R	Performs a read operation in last address read or written
Data Write	Wxxxx:yyyyyyyy	Performs a write operation of $yyyyyyyy data in address $xxxx

Note: All results from reading or writing registers using UART interface are expressed in hexadecimal.

As an example of the operative with this interface, the next command sequence explains how to set a value (0x584E9140 in hexadecimal) into the register address 0x08 of the **ME**S IP core.

```
# READ REGISTER 0x08
> R0008
  Read 00000000 from address 0008

# WRITE 0x584E9140 INTO REGISTER 0x08
> W0008:58489140
  Written 58489140 to address 0008
```

```
# READ REGISTER 0x08
> R0008
  Read 58489140 from address 0008
```

7.2. COE interface

This interface type uses Ethernet as a way to communicate. The COE frames have a preconfigured Ethernet frame format (it depends on the type of operation) which is interpreted by the **MES** IP core (COE module). The access to status and configuration registers is possible from any of the Ethernet ports of th IP core. Note that when this interface type is selected, an extra port, apart from those specified in the *Number of Ports*, is added to the **MES** IP Core where the COE module is connected.

In the figure 7.1 are shown the write/read COE frames and their associated responses. Note that the data has been edited in ASCII characters, but they shall be transmitted in the corresponding hexadecimal value (i.e W 0x57). In the same way, read commands and data are received in the corresponding hexadecimal value.

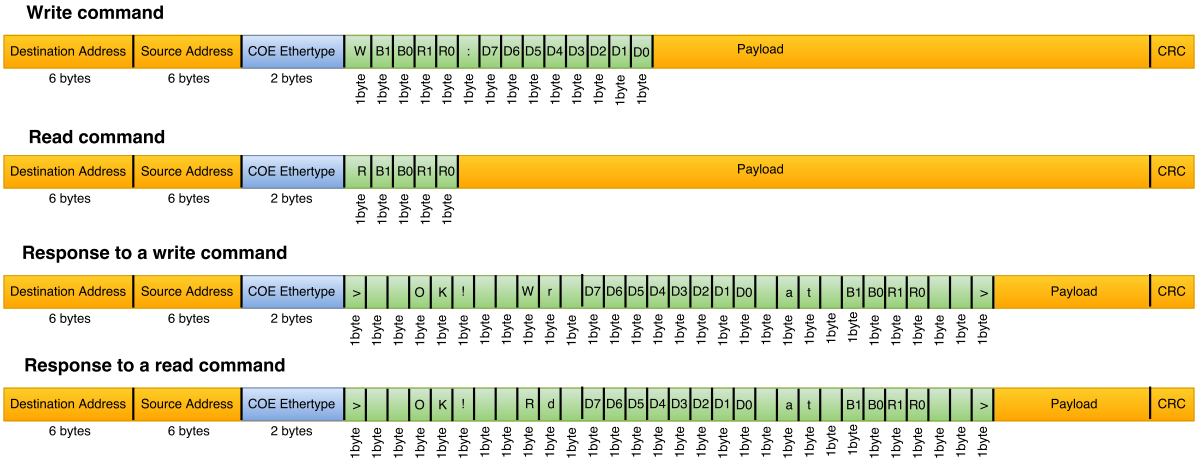


Figure 7.1.: **COE frame formats.**

The nomenclature used in the figure 7.1 is:

- **W**: Write command.
- **R**: Read command.
- B_1B_0: Wishbone slave module address.
- R_1R_0: Register address.

  Note: Absolute address $= B_1B_0R_1R_0$

- $D_7...D_0$: Data to write in a write command or read data in a read command.

Below there is an example of a read command in the address x"0010" (**R0010**):

R 0 0 1 0

dst addr |src addr |ethertype |x"52" |x"30" |x"30" |x"31" |x"30" |payload|fcs

7.3. MDIO interface

ME*S* IP core allows status checking and control via MDIO interface. This serial communication interface is generally used to configure PHY devices. Following the same idea, **ME***S* IP core responds to register read and write using this protocol.

MDIO uses two lines, one for data and another for the clock. All MDIO slaves hang from those lines and a master selects the device to manage by communicating the PHY address of that device through the data line. Each slave has several registers with their own register addresses.

ME*S* IP core internal MDIO slave interface provides a "fake mii table" at 0x1F PHY address that allows direct MAC to MAC connection without PHYs to an external CPU. This MII table's responses depend on the selected speed.

In order to get access to **ME***S* registers, latest MII table registers (28 to 31) are used as follows:

- Register 28 is indirect address register.
- Register 29 is indirect data [31:16] register.
- Register 30 is indirect data [15:0] register.

The correct sequence in order to write core registers is:

- Put address on register 28
- Put higher bits on register 29
- Put lower bits on register 30 (This will perform the write action)

The correct sequence in order to read a **ME***S* core register is:

- Put address on register 28
- Read data on register 30 (This will perform the read action) (These are the bits [15:0] of the core register value)
- Read data on register 29 (These are the bits [31:16] of the core register value)

MES IP core also includes an internal MDIO master that allows to access the ports PHYs. The figure 7.2 shows the **MES** IP core's architecture with the MDIO interfaces.

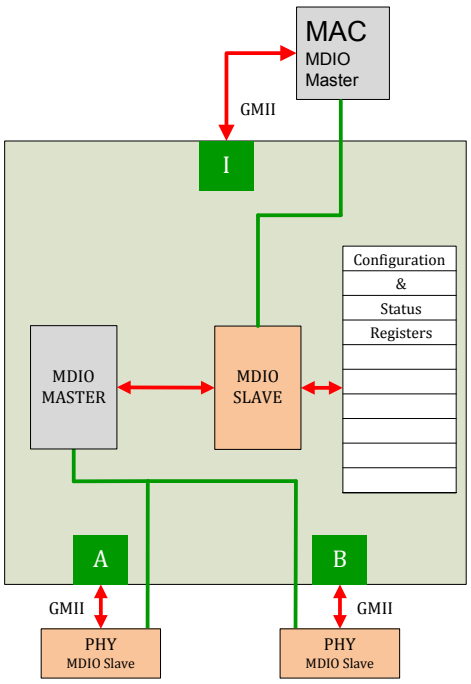


Figure 7.2.: **General MDIO interface connections.**

7.4. AXI4 interface

It allows a point to point, master/slave interface using a defined AXI signalling standard in order to access configuration and statistic registers.

8. Register memory map

When the core is generated with a management interface, all control and status registers are memory mapped. The complete register map has a modular distribution built based on the modules with accessible registers. After power-up or reset, you can reconfigure the core parameters from their defaults at any time.

Each of these modules has its **Base Address**. To access registers inside a module it has to take into account that:

Absolute address = Module Base address + Register Address offset

Table 8.1 shows the distribution of base addresses of the different wishbone slave modules that compose the core. The register list of each module is described in the following sections.

Table 8.1.: **Base addresses of the complete register space.**

Wishbone slave module	Base Address	Description
Register bank	0x0000	Design global parameters
IEEE1588 TC Timer	0x0100	1588 Transparent Clock or PTB Timer configuration
MDIO Bridge	0x0200	MDIO bus accessing
I2C Bridge	0x0300	I2C bus accessing
IP Status	0x0400	IP status signalling
Ethernet Switching Engine (Bank 0)	0x0500	Ethernet Switching Engine configuration
Ethernet Switching Engine (Bank 1)	0x0600	Ethernet Switching Engine configuration
Ethernet Port 0	0x0700	Port and Transparent Clock - P2P configuration
..	...	...
Ethernet Port n	0x0(7+n)00	Port and Transparent Clock - P2P configuration

In case that a non defined register or a register in a non implemented wishbone slave module is accessed, the core automatically response in order to avoid the IP being got stuck. So if an register out of the defined range is accessed, the wishbone slave module responses with *x"DEADDEAD"*. If a register in a non implemented wishbone slave is accessed, *x"XXXXDEAD"* is reported (where XXXX is the accessed wishbone slave number).

NOTE 1: This register memory map as well as its registers could have changed since the edition of this document.

NOTE 2: For more information about how to manage and configure registers, see Appendix A.

8.1. Register Bank

There is global information about the design. A list of the general purpose registers is shown in Table 8.2.

Table 8.2.: **Global registers bank**

Address (Hex)	Description
0x00	Core Version
0x04	MAC Address LO
0x08	MAC Address HI
0x0C	SoC-e Licensed Features
0x10	GUI Enabled Features
0x14	IP Address

This MAC value can change depending on the design

The contents of each configuration register are shown in Tables 8.3 to 8.7.

Table 8.3.: **Core Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x19090000	RO	Version of the current design.

Table 8.4.: **MAC_Address_LO [0x04]**

Bits	Default Value	Type	Description
31:0	0x50CE50CE	R/W	**MAC Address LO**: This MAC Address is used by the IP core as the source address for any outbound frames generated by the IP itself. These bits are the low part of the MAC address, bits 31 to 0.

Table 8.5.: **MAC_Address_HI [0x08]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x000050CE	R/W	**MAC Address HI**: This MAC Address is used by the IP core as the source address for any outbound frames generated by the IP itself. These bits are the high part of the MAC address, bits 47 to 32.

Table 8.6.: **SoC-e Licensed Features [0x0C]**

Bits	Default Value	Type	Description
31:27	0x10	RO	**Maximum Number of Ports**: These bits specify the maximum number of ports enabled in the IP by license.
26:7	N/A	RO	Reserved
6	0x1	RO	**DLR**: This bit specifies if the DLR feature in the IP is enabled or not by license.
5	0x1	RO	**Port mirroring**: This bit specifies if the port mirroring feature in the IP is enabled or not.
4	0x1	RO	**MRP**: This bit specifies if the MRP feature in the IP is enabled or not by license.
3	0x1	RO	**RSTP**: This bit specifies if the RSTP feature in the IP is enabled or not by license.
2	0x1	RO	**Transparent Clock**: This bit specifies if the transparent Clock feature in the IP is enabled or not by license.
1	0x0	RO	**Secure Chip**: This bit specifies if the secure chip checking in the IP is enabled or not by license.
0	0x1	RO	**Time Limitation**: This bit specifies if the time limitation in the IP is enabled or not by license.

Table 8.7.: **GUI Enabled Features [0x10]**

Bits	Default Value	Type	Description
31:20	N/A	RO	Reserved
19	0x0	RO	**Static MAC Table**: This bit specifies if the Static MAC Table is enabled or not in the GUI.
18	0x0	RO	**Multicast Filtering**: This bit specifies if the Multicast Filtering is enabled or not in the GUI.
17	0x0	RO	**Per Port Rate Limiting**: This bit specifies if the Per Port Rate Limiting functionality is enabled or not in the GUI.
16	0x0	RO	**Port Mirroring**: This bit specifies if the port mirroring functionality is enabled or not in the GUI.
15	0x0	RO	**RSTP Support**: This bit specifies if the RSTP protocol is enabled or not in the GUI.
14	0x0	RO	**Jumbo Frame Support**: This bit specifies if the Jumbo frame support is enabled or not in the GUI.
13	0x0	RO	**Priority Support**: This bit specifies if priorities are enabled or not in the GUI.
12:10	0x2	RO	**Priority Buffers**: These bits specify the number of priority buffers set in the GUI.
9:8	0x0	RO	**Ring Protocol**: These bits specify which ring protocol is set in the GUI.
7:5	0x0	RO	**IEEE1588 Support**: These bits specify which IEEE1588 mode is set in the GUI.
4:0	0x5	RO	**Number of Ports**: These bits specify the number of ports set in the GUI.

Table 8.8.: **IP_Address [0x14]**

Bits	Default Value	Type	Description
31:0	0xC0A80164	R/W	**IP Address**: This IP Address is used by the IP core as the IP source address for any outbound frames generated by the IP itself.

8.2. Ethernet Switching Engine

There are Ethernet Switching Engine configuration registers. The banks of register list are shown in Tables 8.9 to 8.11.

Table 8.9.: **Management Ethernet Switch registers (Bank 0) - Table 1**

Address (Hex)	Description
0x00	Module Version
0x04	Forwarding Behaviour Configuration
0x08	Reserved
0x0C	Native VLAN
0x10	VLAN ID mask configuration
0x14	Switching mask
0x18	MAC Table Clear
0x1C	RSTP Port Status 1
0x20	Ageing time
0x24	Reserved
0x28	Priority by Ethertype
0x2C	Forwarding by Ethertype
0x30	Forwarding by Ethertype Control
0x34	Per Port Frame Rate Limiting Control
0x38	MAC address query control
0x3C	MAC address query HI
0x40	MAC address query LO
0x44	Multicast Filtering Control
0x48	Reserved
0x4C	Reserved
0x50	MRP role
0x54	MRP_LNKupT
0x58	MRP_LNKdownT
0x5C	MRP_TS_Prio
0x60	MRP_TSTNRmax
0x64	MRP_TSTdefaultT
0x68	MRP_TSTshortT
0x6C	MRP_TOPchgT
0x70	MRP_TSTExtNRmax
0x74	Port mirroring configuration
0x78	Static MAC address LO
0x7C	Static MAC address HI and control
0x80	DLR Control
0x84	DLR Network topology
0x88	DLR Network status
0x8C	DLR Ring supervisor status (*)
0x90	DLR Ring supervisor enable (*)
0x94	DLR Ring supervisor precedence (*)
NOTE: (*) These registers are valid for devices capable of functioning as a ring supervisor.	

Table 8.10.: **Management Ethernet Switch registers (Bank 0) - Table 2**

Address (Hex)	Description
0x98	DLR Beacon interval (*)
0x9C	DLR Beacon timeout (*)
0xA0	DLR VLAN ID
0xA4	Reserved
0xA8	DLR Ring faults count (*)
0xAC	DLR Last active node on port 1 (MAC address HI) (*)
0xB0	DLR Last active node on port 1 (MAC address LO) (*)
0xB4	DLR Last active node on port 1 (IP address) (*)
0xB8	DLR Last active node on port 2 (MAC address HI) (*)
0xBC	DLR Last active node on port 2 (MAC address LO) (*)
0xC0	DLR Last active node on port 2 (IP address) (*)
0xC4	DLR Ring protocol participants count (*)
0xC8	DLR Active supervisor address (MAC address HI)
0xCC	DLR Active supervisor address (MAC address LO)
0xD0	DLR Active supervisor IP address
0xD4	DLR Active supervisor precedence (*)
0xD8	DLR Capability flags
0xDC	DLR Services (*)
0xE0	VLAN ID mask
0xE4	RSTP Port Status 2
0xE8	Static MAC origin ports
0xEC	Broadcast Frame Rate limit
0xF0	Multicast Frame Rate limit
0xF4	Unicast Frame Rate limit
0xF8	Per Port Throughput Limiting Control
0xFC	IGMP Smooping
NOTE: (*) These registers are valid for devices capable of functioning as a ring supervisor.	

The contents of each configuration register are shown in Tables 8.12 to 8.84.

Table 8.11.: **Management Ethernet Switch registers (Bank 1)**

Address (Hex)	Description
0x00	DSA MAC Address 1 HI
0x04	DSA MAC Address 1 LO
0x08	DSA MAC Address 2 HI
0x0C	DSA MAC Address 2 LO
0x10	DSA MAC Address 3 HI
0x14	DSA MAC Address 3 LO
0x18	DSA MAC Address 4 HI
0x1C	DSA MAC Address 4 LO
0x20	VLAN Intra Ports Flooding
0x24	Mirrored Ports
0x28	8021X Port Control
0x2C	8021X Configuration
0x30	8021X MAC Authentication Low
0x34	8021X MAC Authentication High

Table 8.12.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x19090000	RO	**Module Version**: Version number for internal use.

Table 8.13.: **Forwarding behaviour configuration [0x04]**

Bits	Default Value	Type	Description
31	0x0	R/W	**Port-based Virtual VLAN**: When this bit is set to 1, port-based Virtual VLAN functionality is enabled.
30	0x0	R/W	**Switching Portmask**: When this bit is set to 1, Switching port mask functionality is enabled. It is possible to enable both *Port-based Virtual* and *Switching Portmask* at the same time. In this case, the final decision of the output forwarding port(s) is based on the combination of both functionalities.
29:(n ports)	N/A	RO	Reserved.
(n ports-1):0	0x0	R/W	**Port mode selector**: When *Port-based Virtual VLAN* bit is enabled, these bits specify the mode in which each port works (each bit represents each port). If 1, the port is in trunk mode. If 0, the port is in access mode.

Table 8.14.: **Native VLAN [0x0C]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits specify the port in which a read or write process is performed.
26:17	N/A	RO	Reserved.
16:14	0x0	R/W	**VLAN PCP**: When *Wr/Rd* is set to 1, it is the Priority code point (PCP) value to write. When *Wr/Rd* is set to 0, it is the read PCP value.
13	0x0	R/W	**VLAN CFI**: When *Wr/Rd* is set to 1, it is the Canonical Format Indicator (CFI) value to write. When *Wr/Rd* is set to 0, it is the read VLAN ID value.
12:1	0x0	R/W	**VLAN ID**: When *Wr/Rd* is set to 1, it is the VLAN ID value to write. When *Wr/Rd* is set to 0, it is the read VLAN ID value.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.15.: **VLAN ID mask configuration [0x10]**

Bits	Default Value	Type	Description
31:20	0x0	R/W	**VLAN ID**: These bits specify the VLAN ID in which the VLAN mask is assigned or the VLAN ID over which the VLAN mask is consulted.
19:2	N/A	RO	Reserved.
1	0x0	WO	**Clear VLAN mask**: This bit is used to clear all configuration regarding VLAN mask.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.16.: **Switching mask [0x14]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits specify the port in which a read or write process is performed.
26:(n ports+1)	N/A	RO	Reserved.
(n ports):1	0xFFFF	R/W	**Ports mask**: These bits are used to specify the ports through which a frame received in a specific port could be transmitted. Each bit corresponds to each port and '1' is assigned if a frame can go through this port, '0' otherwise. Note that the width of this field depends on the number of ports.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.17.: **MAC Table Clear [0x18]**

Bits	Default Value	Type	Description
31:2	N/A	RO	Reserved.
1	0x0	R/W	**Static MAC Table Flush**: When *Static MAC Table* is enabled and if is set to 1, the static MAC table is flushed. If 0, nothing is done.
0	0x0	R/W	**Dynamic MAC Table Flush**: If it is set to 1, the dynamic MAC table is flushed. If 0, nothing is done.

Table 8.18.: **RSTP Port Status 1 [0x1C]**

Bits	Default Value	Type	Description
31:30	0x3	R/W	**Port 15 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
...	...	...	**Port 14 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
3:2	0x3	R/W	**Port 1 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
1:0	0x3	R/W	**Port 0 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding

*NOTE 1: When RSTP is enabled, the management port is by default in **forwarding** mode whereas other ports are in **discarding** mode.*
NOTE 2: This register is used with the first 16 ports.

Table 8.19.: **Ageing time [0x20]**

Bits	Default Value	Type	Description
31:10	N/A	RO	Reserved.
9:0	0x3C	R/W	**Aging Time**: Specifies the aging time in seconds for dynamically learned address. Possible values goes from 0 to 1024 seconds. Note that aging time 0 means that there is no aging process. The aging time granularity is $aging\_time/2$. For example, if $aging\_time = 60$ seconds, a MAC entry could last between 60 and 90 seconds before becoming invalid.

Table 8.20.: **Priority by Ethertype [0x28]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits are used to select the port in which the priority mode and its configuration are specified.
26:24	N/A	RO	Reserved.
23	0x0	RO	**Not valid priority**: This bit says if the read priority associated to the Ethertype is valid or not. If '1', the consulted Ethertype is not among the configured ones.
22:20	0x0	R/W	**Priority**: These bits are used to set the priority buffer associated to the configured Ethertype.
19:4	0x0	R/W	**Ethertype**: These bits are used to set the Ethertype.
3:2	0x0	R/W	**Priority type**: These bits are used to select the QoS scheme. - **00:** PCP bits based priority (by default). - **01:** DSCP TOS bits based priority. - **10:** Ethertype based priority.
1	0x0	WO	**Clear Ethertype based priority**: When *Priority type* is set "10" and if it is set to 1, it clears all previously configured priorities based on Ethertype.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.21.: **Forwarding by Ethertype [0x2C]**

Bits	Default Value	Type	Description
31	0x0	RO	**Not valid Ethertype entry**: This bit says if the result of the consultation of the Ethertype is valid or not. If '1', the consulted Ethertype is not among the configured ones.
30:(*n* ports)	N/A	RO	Reserved
(*n* ports-1):0	0x0	R/W	**Forwarding ports**: When *Forwarding by Ethertype active* is set to 1 (Table 8.22), these bits specify the ports through which frames with the specified Ethertype can be transmitted (each bit corresponds to each port). Note that the width of this field depends on the number of ports. On the other hand, when a consultation of the output ports of an Ethertype is performed, its result is shown in this register.

Table 8.22.: **Forwarding by Ethertype control [0x30]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits are used to select the port in which the forwarding behaviour based on the Ethertype is set.
26:24	N/A	RO	Reserved.
23:8	0x0	R/W	**Ethertype**: This value specifies the Ethertype to be read/written.
7:3	N/A	RO	Reserved.
2	0x0	WO	**Clear Ethertype based forwarding**: If 1, it clears all previously configured forwarding behaviour based on Ethertype. If 0, it does nothing.
1	0x0	R/W	**Forwarding by Ethertype active**: If 1, Ethertype based forwarding is enabled. If 0, Ethertype based forwarding is disabled.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.23.: **Per Port Frame Rate Limiting Control [0x34]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits are used to select the port in which Frame Rate Limiting will be applied.
26:4	N/A	RO	Reserved.
3	0x0	R/W	**Broadcast frame rate limiting enable**: This bit is used to enable broadcast frames rate limiting.
2	0x0	R/W	**Multicast frame rate limiting enable**: This bit is used to enable multicast frames rate limiting.
1	0x0	R/W	**Unicast frame rate limiting enable**: This bit is used to enable unicast frames rate limiting.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.24.: **MAC address query control [0x38]**

Bits	Default Value	Type	Description
31:9	N/A	RO	Reserved.
8:4	0x0	R/W	**Port selector**: These bits are used to select the port in which a consultation is done.
3	0x0	R/W	**MAC Table select**: This bit is used to select over which MAC table, dynamic ('0') or static ('1'), the consultation is done.
2	0x0	WO	**MAC query cancel**: If it is set to 1, it cancels the current consultation.
1	0x0	WO	**MAC query continue**: If it is set to 1, it enables to continue with the searching process for the next MAC address.
0	0x0	WO	**MAC query start**: If it is set to 1, a new consultation starts. Note that if this bit is asserted while another consultation is under service, a new consultation is started discarding the previous one.

Table 8.25.: **MAC address query HI [0x3C]**

Bits	Default Value	Type	Description
31:16	0x0	RO	**MAC address query HI**: These bits show the read MAC address as a result of a consultation, bits 47 to 32.
15:7	N/A	RO	Reserved.
6:2	0x0	RO	**Port selector**: These bits specify the port in which its MAC addresses are consulted.
1	0x0	RO	**MAC address consultation valid**: If 1, it indicates that the read MAC address is valid.
0	0x0	RO	**MAC address consultation finished**: If 1, it indicates that the consultation process has finished.

Table 8.26.: **MAC address query LO [0x40]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**MAC address query LO**: These bits show the read MAC address as a result of a consultation, bits 31 to 0.

Table 8.27.: **Multicast Filtering Control [0x44]**

Bits	Default Value	Type	Description
31:1	N/A	RO	Reserved.
2	0x0	R/W	**Unregistered multicast packet flooding**: If '1', it enables the flooding of unregistered multicast packets. It is used fro IGMP snooping functionality.
1	0x0	R/W	**Multicast ip routing**: If '1', it enables the forwarding of IGMP frames without taking into account any VLAN configuration. It is used fro IGMP snooping functionality.
0	0x0	R/W	**Multicast filter enable**: If '1', it enables the multicast filtering functionality.

Table 8.28.: **MRP role [0x50]**

Bits	Default Value	Type	Description
31:3	N/A	RO	Reserved
2	0x0	R/W	**MRP enable**: If 1, it enables MRP functionality. If 0, MRP functionality is disabled.
1	0x0	R/W	**MRM**: If 1, it enables Media Redundancy Manager (MRM) role.
0	0x1	R/W	**MRC**: If 1, it enables Media Redundancy Client (MRC) role. It is the default role.

NOTE: It is not possible to enable both MRM and MRC at the same time.

Table 8.29.: **MRP_LNKupT [0x54]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x1	R/W	**MRP LNKupT**: These bits specify the Link Up Timer interval (in **ms**).

Table 8.30.: **MRP_LNKdownT [0x58]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x1	R/W	**MRP LNKdownT**: These bits specify the Link Down Timer interval (in **ms**).

Table 8.31.: **MRP_TS_Prio [0x5C]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x0	R/W	**MRP_TS_Prio**: These bits specify the priority of the host.

Table 8.32.: **MRP_TSTNRmax [0x60]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x5	R/W	**MRP_TSTNRmax**: These bits specify the MRP_Test monitoring count (in **ms**).

Table 8.33.: **MRP_TSTdefaultT [0x64]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x32	R/W	**MRP_TSTdefaultT**: These bits specify the MRP_Test default interval (in **ms**).

Table 8.34.: **MRP_TSTshortT [0x68]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x1E	R/W	**MRP_TSTshortT**: These bits specify the MRP_Test short interval (in **ms**).

Table 8.35.: **MRP_TOPchgT [0x6C]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x14	R/W	**MRP_TOPchgT**: These bits specify the Topology Change (Clear Address Table) request interval (in **ms**).

Table 8.36.: **MRP_TSTExtNRmax [0x70]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0xF	R/W	**MRP_TSTExtNRmax**: These bits specify the MRP_Test extended monitoring count.

Table 8.37.: **Port mirroring configuration [0x74]**

Bits	Default Value	Type	Description
31:6	N/A	RO	Reserved
5:1	0x0	R/W	**Mirror port**: This specifies which port is the mirror port.
0	0x0	R/W	**Port mirroring enable**: If 1, it enables port mirroring functionality.

Table 8.38.: **Static MAC address LO [0x78]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Static MAC Address LO**: These bits show the low part of the written Static MAC address, bits 31 to 0

Table 8.39.: **Static MAC address HI and control [0x7C]**

Bits	Default Value	Type	Description
31:18	N/A	RO	Reserved
17	0x0	WO	**Static source MAC address clear**: If 1, it removes the specified Static MAC address from the Static MAC table.
16	0x0	WO	**Static source MAC address write**: If 1, it commands a write operation of the specified Static MAC address.
15:0	0x0	R/W	**Static MAC Address HI**: These bits show the high part of the written Static MAC address, bits 47 to 32

Table 8.40.: **DLR control [0x80]**

Bits	Default Value	Type	Description
31:2	N/A	RO	Reserved
1	0x0	R/W	**DLR functionality enable**: This bit is used to enable DLR functionality. Disabled by default.
0	Depends on the GUI configuration	R/W	**Ring Node Type**: This bit defines the DLR node type: 0 - Beacon based ring node 1 - Ring supervisor

Table 8.41.: **DLR Network topology [0x84]**

Bits	Default Value	Type	Description
31:1	N/A	RO	Reserved
0	0x0	RO	**Current Network Topology Mode**: This bit defines the current network topology mode: 0 - Linear 1 - Ring

Table 8.42.: **DLR Network status [0x88]**

Bits	Default Value	Type	Description
31:4	N/A	RO	Reserved
3:0	0x0	RO	**Current Status of Network**: These bits define the current status of network: 0 - Normal operation in both, Ring and Linear Network Topology modes 1 - Ring Fault. A ring fault has been detected. Valid only when Network Topology is Ring 2 - Unexpected loop detected 3 - Partial Network Fault. A network fault has been detected in one direction only. Valid only when Network Topology is Ring and the node is the active ring supervisor 4 - Rapid Fault/Restore Cycle. A series of rapid ring fault/restore cycles has been detected. The condition shall be cleared explicitly via the “Clear Rapid Faults” service

Table 8.43.: **DLR Ring supervisor status [0x8C]**

Bits	Default Value	Type	Description
31:4	N/A	RO	Reserved
3:0	0x0	RO	**Ring Supervisor Active Status Flag**: these bits indicate the device's status as a ring supervisor. 0 - The device is functioning as a backup supervisor 1 - The device is functioning as the active ring supervisor 2 - The device is functioning as a normal ring node (supervisor not enabled) 3 - The device is functioning in a non-DLR topology (supervisor not enabled, and no other supervisor is present) 4 - The device cannot support the currently operating ring parameters (Beacon Interval and/or Beacon Timeout)

Table 8.44.: **DLR Ring supervisor enable [0x90]**

Bits	Default Value	Type	Description
31:4	N/A	RO	Reserved
0	0x0	R/W	**Ring Supervisor Enable Flag**: this bit is used to determine if the node is configured as ring supervisor node or normal ring node: 0 - The device is configured as a normal ring node (default) 1 - The device is configured as a ring supervisor

Table 8.45.: **DLR Supervisor Precedence [0x94]**

Bits	Default Value	Type	Description
31:8	N/A	RO	Reserved
7:0	0x0	R/W	**Precedence**: Precedence of a ring supervisor in network with multiple ring supervisors. Numerically higher value indicates higher precedence. Default value is 0.

Table 8.46.: **DLR Beacon interval [0x98]**

Bits	Default Value	Type	Description
31:0	0x190	R/W	**Beacon Interval**: Duration of ring beacon interval in μs units). The default value is 400 μs. When it works as an backup supervisor, it shows the Beacon Interval value of the Active Supervisor node.

Table 8.47.: **DLR Beacon timeout [0x9C]**

Bits	Default Value	Type	Description
31:0	0x7A8	R/W	**Beacon timeout**: Amount of time nodes (in μs units) shall wait before timing out reception of Beacon frames and taking the appropriate action. The default value is 1960 μs. When it works as an backup supervisor, it shows the Beacon timeout value of the Active Supervisor node.

Table 8.48.: **DLR VLAN ID [0xA0]**

Bits	Default Value	Type	Description
31:12	N/A	RO	Reserved
11:0	0x0	R/W	**DLR Ring VLAN ID**: These bits determine the VLAN ID used in ring protocol frames. Default value is 0. When it works as an backup supervisor, it shows the VLAN ID value of the Active Supervisor node.

Table 8.49.: **DLR Ring faults count [0xA8]**

Bits	Default Value	Type	Description
31:17	N/A	RO	Reserved
16	0x0	WO	**Reset Ring Faults Counter**: This bit is used to reset the Ring Faults Counter.
15:0	0x0	RO	**Ring Faults Count**: It contains the number of times since power up that the device has detected a ring fault, as either active or backup supervisor. If the Ring Supervisor Enable is set to FALSE, the Ring Faults Count shall be set to 0.

Table 8.50.: **DLR Last active node on port 1 (MAC address HI) [0xAC]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x0	RO	**MAC Address Last active node on port 1 (HI)**: Ethernet MAC address of the last node reachable through port 1 of an active ring supervisor. These bits show the high part of the MAC address, bits 47 to 32.

Table 8.51.: **DLR Last active node on port 1 (MAC address LO) [0xB0]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**MAC Address Last active node on port 1 (LO)**: Ethernet MAC address of the last node reachable through port 1 of an active ring supervisor. These bits show the low part of the MAC address, bits 31 to 0.

Table 8.52.: **DLR Last active node on port 1 (IP address) [0xB4]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**IP Address Last active node on port 1**: IP address of the last node reachable through port 1 of an active ring supervisor.

Table 8.53.: **DLR Last active node on port 2 (MAC address HI) [0xB8]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x0	RO	**MAC Address Last active node on port 2 (HI)**: Ethernet MAC address of the last node reachable through port 2 of an active ring supervisor. These bits show the high part of the MAC address, bits 47 to 32.

Table 8.54.: **DLR Last active node on port 2 (MAC address LO) [0xBC]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**MAC Address Last active node on port 2 (LO)**: Ethernet MAC address of the last node reachable through port 2 of an active ring supervisor. These bits show the low part of the MAC address, bits 31 to 0.

Table 8.55.: **DLR Last active node on port 2 (IP address) [0xC0]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**IP Address Last active node on port 2**: IP address of the last node reachable through port 2 of an active ring supervisor.

Table 8.56.: **DLR Ring protocol participants count [0xC4]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x0	RO	**Ring Protocol Participants Count**: this contains the number of members in the Ring Protocol Participants List attribute. The count is gathered by the active ring supervisor through *Sign_On* frame.

Table 8.57.: **DLR Active supervisor address (MAC address HI) [0xC8]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved
15:0	0x0	RO	**Active supervisor address (HI)**: it contains the Ethernet MAC address of the active ring supervisor. The initial values of Ethernet MAC address is 0, until the active ring supervisor is determined. These bits show the high part of the MAC address, bits 47 to 32.

Table 8.58.: **DLR Active supervisor address (MAC address LO) [0xCC]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Active supervisor address (LO)**: it contains the Ethernet MAC address of the active ring supervisor. The initial value of Ethernet MAC address is 0, until the active ring supervisor is determined. These bits show the low part of the MAC address, bits 31 to 0.

Table 8.59.: **DLR Active supervisor IP address [0xD0]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Active supervisor IP address**: it contains the IP address of the active ring supervisor. The initial value of the IP address is 0, until the active ring supervisor is determined. .

Table 8.60.: **DLR Active supervisor precedence [0xD4]**

Bits	Default Value	Type	Description
31:8	N/A	RO	Reserved
7:0	0x0	RO	**Active Supervisor Precedence**: it contains the precedence value of the active ring supervisor. The initial value is 0, until the active ring supervisor is determined.

Table 8.61.: **DLR Capability flags [0xD8]**

Bits	Default Value	Type	Description
31:8	N/A	RO	Reserved
7	0x0	RO	***Flush_Table* frame Capable**: It is set if device is capable of supporting the *Flush_Tables* frame. Always '1'.
6	0x0	N/A	**Redundant Gateway Capable**: It is set if device is capable of providing the redundant gateway function. Always '0'.
5	0x0	RO	**Supervisor Capable**: It is set if device is capable of providing the supervisor function.
4:2	N/A	RO	Reserved
1	0x0	RO	**Beacon-based Ring Node**: It is set if device's ring node implementation is based on processing of Beacon frames.
0	0x0	N/A	**Announce-based Ring Node**: It is set if device's ring node implementation is based on processing of Announce frames. Always '0'.

Table 8.62.: **DLR Services [0xDC]**

Bits	Default Value	Type	Description
31:8	N/A	RO	Reserved
7:4	0x0	RO	**Object State Conflict**: These bits are used to report the status code. If there is an error, it reports 0x0C.
3	0x0	WO	**Reset *Object State Conflict***: This bit is used to restart *Object State Conflict* value.
2	0x0	WO	***Restart_Sign_On***: This bit is used to restart Sign On process.
1	0x0	WO	***Clear_Rapid_Faults***: This bit is used to clear the Rapid Fault/Restore Cycle Detected condition in the ring supervisor, allowing the supervisor to return to normal operation.
0	0x0	WO	***Verify_Fault_Location***: This bit is used to cause ring supervisor to verify fault location by issuing *Locate_Fault* ring protocol message to ring nodes and update Last Active Node 1 and Last Active Node 2 attributes.

Table 8.63.: **VLAN ID mask [0xE0]**

Bits	Default Value	Type	Description
31:(*n* ports)	N/A	RO	Reserved
(*n* ports-1):0	0x0	R/W	**Ports in VLAN**: When configuring the VLAN mask, these bits are used to specify which ports belong to the specified VLAN ID. Each bit corresponds to each port and '1' is assigned if the port belongs to that VLAN ID, '0' otherwise. Note that the width of this field depends on the number of ports. On the other hand, when a consultation of the VLAN mask is performed, its result is shown in this register.

Table 8.64.: **RSTP Port Status 2 [0xE4]**

Bits	Default Value	Type	Description
31:30	0x3	R/W	**Port n state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
...	...	...	**Port n-1 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
3:2	0x3	R/W	**Port 17 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding
1:0	0x3	R/W	**Port 16 state**: It defines port status. **00** - Discarding **01** - Learning **11** - Forwarding

*NOTE 1: When RSTP is enabled, the management port is by default in **forwarding** mode whereas other ports are in **discarding** mode.*

NOTE 2: This register is used when the number of ports is greater than 16.

Table 8.65.: **Static MAC origin ports [0xE8]**

Bits	Default Value	Type	Description
31:(*n* ports)	N/A	RO	Reserved
(*n* ports-1):0	0x0	R/W	**Origin Ports of Static MAC Address**: These bits specify the origin ports of the Static MAC address. Note that one static MAC address can have associated more than one origin port, for example, when a multicast MAC address is configured.

Table 8.66.: **Broadcast Frame Rate limit [0xEC]**

Bits	Default Value	Type	Description
31:17	N/A	RO	Reserved.
15:0	0x0	R/W	**Broadcast Frame max limit**: This value is the maximum number of broadcast frames allowed in the configured window. The check windows depends on the PHY speed and are: - 10 Mbps: 100 ms - 100 Mbps: 10 ms - 1000 Mbps: 1 ms

Table 8.67.: **Multicast Frame Rate limit [0xF0]**

Bits	Default Value	Type	Description
31:17	N/A	RO	Reserved.
15:0	0x0	R/W	**Multicast Frame max limit**: This value is the maximum number of multicast frames allowed in the configured window. The check windows depends on the PHY speed and are: - 10 Mbps: 100 ms - 100 Mbps: 10 ms - 1000 Mbps: 1 ms

Table 8.68.: **Unicast Frame Rate limit [0xF4]**

Bits	Default Value	Type	Description
31:17	N/A	RO	Reserved.
15:0	0x0	R/W	**Unicast Frame max limit**: This value is the maximum number of unicast frames allowed in the configured window. The check windows depends on the PHY speed and are: - 10 Mbps: 100 ms - 100 Mbps: 10 ms - 1000 Mbps: 1 ms

Table 8.69.: **Per Port Throughput Limiting Control [0xF8]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**Port selector**: These bits are used to select the port in which throughput limiting configuration will be applied.
26:19	N/A	RO	Reserved.
18:2	0x0	R/W	**Throughput max limit**: This value is the maximum number of bytes allowed in the configured window. The check windows depends on the PHY speed and are: - 10 Mbps: 100 ms - 100 Mbps: 10 ms - 1000 Mbps: 1 ms
1	0x0	R/W	**Throughput limiting enable**: This bit is used to enable unicast frames rate limiting.
0	0x0	WO	**Wr/Rd**: If 1, write process is commanded. If 0, read process.

Table 8.70.: **IGMP Snooping [0xFC]**

Bits	Default Value	Type	Description
31	0x0	R/W	**IGMP Snooping Enable**: This bit enables IGMP Snooping functionality..
30:n^o *ports*	N/A	RO	Reserved.
(n^o *ports*-1):0	0x0	R/W	**Router Ports**: These bits specify which ports are connected to a router.

Table 8.71.: **DSA MAC Address 1 HI [0x00]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15:0	0x0	R/W	**DSA MAC 1 Address HI**: These bits show the most significant bits of the first MAC address to be tagged with DSA, bits 47 to 32.

Table 8.72.: **DSA MAC Address 1 LO [0x04]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**DSA MAC 1 Address LO**: These bits show the less significant bits of the first MAC address to be tagged with DSA, bits 31 to 0.

Table 8.73.: **DSA MAC Address 2 HI [0x08]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15:0	0x0	R/W	**DSA MAC 2 Address HI**: These bits show the most significant bits of the second MAC address to be tagged with DSA, bits 47 to 32.

Table 8.74.: **DSA MAC Address 2 LO [0x0C]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**DSA MAC 2 Address LO**: These bits show the less significant bits of the second MAC address to be tagged with DSA, bits 31 to 0.

Table 8.75.: **DSA MAC Address 3 HI [0x10]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15:0	0x0	R/W	**DSA MAC 3 Address HI**: These bits show the most significant bits of the third MAC address to be tagged with DSA, bits 47 to 32.

Table 8.76.: **DSA MAC Address 3 LO [0x14]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**DSA MAC 3 Address LO**: These bits show the less significant bits of the third MAC address to be tagged with DSA, bits 31 to 0.

Table 8.77.: **DSA MAC Address 4 HI [0x18]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15:0	0x0	R/W	**DSA MAC 4 Address HI**: These bits show the most significant bits of the fourth MAC address to be tagged with DSA, bits 47 to 32.

Table 8.78.: **DSA MAC Address 4 LO [0x1C]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**DSA MAC 4 Address LO**: These bits show the less significant bits of the fourth MAC address to be tagged with DSA, bits 31 to 0.

Table 8.79.: **VLAN Intra Ports Flooding [0x20]**

Bits	Default Value	Type	Description
31:1	N/A	RO	Reserved.
0	0x0	R/W	**VLAN Intra Ports Flooding**: This bit is used to enable a non standard frame forwarding behaviour. When it is enabled and the resulting destination ports value after combining the MAC and VLAN Tables is zero, frame is flooded towards all ports that belong to the same VLAN than the origin frame.

Table 8.80.: **Mirrored Ports [0x24]**

Bits	Default Value	Type	Description
31:(n ports)	N/A	RO	Reserved
(n ports - 1):0	0x0	R/W	**Mirror origin port**: These bits define which ports are the origin ports to be mirrored. Each bit corresponds to each port. Note that the width of this field depends on the number of ports.

Table 8.81.: **8021X Port Control [0x28]**

Bits	Default Value	Type	Description
31:9	0x0	RO	Reserved.
8:4	0x0	R/W	**Port ID**: This field defines in which port 802.1X configuration is going to be set or read.
0	0x0	R/W	**Wr enable**: If 1, 802.1X configuration is stored for the specified port.

Table 8.82.: **8021X Configuration [0x2C]**

Bits	Default Value	Type	Description
31:2	0x0	RO	Reserved.
1	0x0	R/W	**MAC address block enable**: If 1 and *IEEE8021.x* is set to true, this enables MAC address based port blocking. If 0, MAC address based port blocking is disabled.
0	0x1	R/W	**Port block enable**: If 1 and *IEEE8021.x* is set to true, this enables port blocking. If 0, port blocking is disabled.

Table 8.83.: **8021X MAC Authentication Low [0x30]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Valid MAC address LO**: It specifies the valid MAC address for 802.1X functionality. These bits are the low part of the MAC address, bits 31 to 0.

Table 8.84.: **8021X MAC Authentication High [0x34]**

Bits	Default Value	Type	Description
31:16	0x0	RO	Reserved.
15:0	0x0	R/W	**Valid MAC address HI**: It specifies the valid MAC address for 802.1X functionality. These bits are the low part of the MAC address, bits 47 to 32.

8.3. IEEE1588 Timer

There is information related with the timer used by the Transparent Clock module. A list of the registers is shown in Table 8.85.

Table 8.85.: **IEEE1588 Timer registers**

Address (Hex)	Description
0x00	Module Version
0x04	Timer Addend
0x08	Timer Period

The contents of each configuration register are shown in Tables 8.86 to 8.88.

Table 8.86.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x17100000	RO	**Module Version**: Version number for internal use.

Table 8.87.: **Timer Addend [0x04]**

Bits	Default Value	Type	Description
31:0	0xE38E38E3	R/W	**Timer addend**: subnanosecond adjustment. See Freescale AN3423 for more information. The fixed default value is for a 125 MHz clock frequency. For 100 MHz clock frequency, the value should be 0xE8BA2E8B.

Table 8.88.: **Timer Period [0x08]**

Bits	Default Value	Type	Description
31:8	N/A	RO	Reserved.
7:0	0x9	R/W	**Timer period**: ns value relating to the period of the timer. See Freescale AN3423 for more information. The fixed default value is for a 125 MHz clock frequency. For 100 MHz clock frequency, the value should be 0xB.

8.4. MDIO Bridge

This register is used to write or read to/from the MDIO external buses. A list of the registers is shown in Table 8.89.

Table 8.89.: **MDIO Bridge register**

Address (Hex)	Description
0x00	Module Version
0x04	MDIO Control

The contents of each configuration register are shown in Tables 8.90 to 8.91.

Table 8.90.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x17010000	RO	**Module Version**: Version number for internal use.

Table 8.91.: **MDIO Control [0x04]**

Bits	Default Value	Type	Description
31:27	0x0	R/W	**PHY Address**: These bits provide the PHY address which is accessed.
26:22	0x0	R/W	**Register Address**: These bits specify the register address which is accessed.
21	0x0	R/W	**Operation**: If 1, then a mdio read operation is performed. If 0, it is a mdio write operation.
20	0x0	R/W	**Operation status**: It specifies the state of the operation (Ready / Start-Operation in progress). If 1, there is an operation in progress. If 0, it is ready for a new operation. Set to 1 to start a new operation.
19:16	0x0	R/W	**MDIO bus selection**: These bits specify the port number over which MDIO operation is performed.
15:0	0x0	R/W	**Data wr/rd**: When there is a write operation, this specifies the data to be written. When there is a read operation, it shows read data.

8.5. I2C Bridge

This register is used to write or read to/from the I2C external buses. A list of the registers is shown in Table 8.92.

Table 8.92.: **I2C Bridge register**

Address (Hex)	Description
0x00	Module Version
0x04	I2C Master Control
0x08	Reserved
0x0C	I2C Read/Write Data LO
0x10	I2C Read/Write Data HI

The contents of each configuration register are shown in Tables 8.93 to 8.96.

Table 8.93.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x17010000	RO	**Module Version**: Version number for internal use.

Table 8.94.: **I2C Master Control [0x04]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15	0x0	R/W	**Operation**: If 1, then a I2C read operation is performed. If 0, it is a I2C write operation.
14:12	0x0	R/W	**Number of Bytes**: This specifies the number of bytes to be read or written.
11:8	0x0	R/W	**I2C bus selection**: These bits are used to select I2C bus output(up to four I2C buses can be addressed.
7:1	0x0	R/W	**Address**: These bits specify the address which is accessed.
0	0x0	R/W	**Operation status**: It specifies the state of the operation (Ready / Start / Operation in progress). If 1, there is an operation in progress. If 0, it is ready for a new operation. Set to 1 to start a new operation.

Table 8.95.: **I2C Read/Write Data LO [0x0C]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Read/Write Data HI**: This is the low part of data to be written or read data, bits 31 to 0.

Table 8.96.: **I2C Read/Write Data HI [0x10]**

Bits	Default Value	Type	Description
31:16	N/A	RO	Reserved.
15:0	0x0	R/W	**Read/Write Data HI**: This is the high part of data to be written or read data, bits 47 to 32.

8.6. IP Status

There is an information about different events that has happened in the IP core. A list of the registers is shown in Table 8.97.

Table 8.97.: **IP Status register**

Address (Hex)	Description
0x00	Module Version
0x04	Interrupt vector
0x08	General status bit mask
0x0C	Port status bit mask
0x10	General status vector
0x14 - 0x50	Port n status vector

The contents of each configuration register are shown in Tables 8.98 to 8.103.

Table 8.98.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x18010000	RO	**Module Version**: Version number for internal use.

Table 8.99.: **Interrupt vector [0x04]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Interrupt vector**: This register notifies about any event indication in the 8.102 and 8.103 registers. When any of the bits of this register is asserted, the IP core notifies an interrupt event. Once any of its bits is asserted, the user is in charge of deasserting it writing '0' in the corresponding bit position.
NOTE: Bit 0 of the "Interrupt vector" is related with the "General status vector". When any of the bits of the "General status vector" is asserted, the Bit 0 of the "Interrupt vector" is also asserted. Bits 1 to n, corresponds to each "Port status vector".			

Table 8.100.: **General status bit mask [0x08]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Port mask**: This register is used to specify the mask to be applied in the event notification of the 8.102 register. Only those bits of the 8.102 register that its bit mask is set will be asserted.

Table 8.101.: **Port status bit mask [0x0C]**

Bits	Default Value	Type	Description
31:0	0x0	R/W	**Port mask**: This register is used to specify the mask to be applied in the event notification of the 8.103 register. Only those bits of the 8.103 register that its bit mask is set will be asserted.
NOTE: This bit mask is applied to all "Port status vector" registers (8.103).			

Table 8.102.: **General status vector [0x10]**

Bits	Default Value	Type	Description
31:3	0x0	RO	Reserved.
2	0x0	RO	**General status vector - Bit 2**: Indicates that DLR Ring is in fault state.
1	0x0	RO	**General status vector - Bit 1**: Indicates that MRP Ring is in fault state.
0	0x0	RO	**General status vector - Bit 0**: Indicates that there has been a frame loop.
NOTE: Each status bit is asserted when the corresponding event has happened and the corresponding mask bit is '1'.			

Table 8.103.: **Port n status vector [0x14 - 0x50]**

Bits	Default Value	Type	Description
31:6	0x0	RO	Reserved.
5	0x0	RO	**Port status vector - Bit 5**: Indicates that no valid Ethernet speed has been measured.
4	0x0	RO	**Port status vector - Bit 4**: Indicates that that an overlength frame has been received.
3	0x0	RO	**Port status vector - Bit 3**: Indicates that that an underlength frame has been received.
2	0x0	RO	**Port status vector - Bit 2**: Indicates that at least one frame was dropped due to congestion in egress buffer.
1	0x0	RO	**Port status vector - Bit 1**: Indicates that at least one frame was dropped due to congestion in ingress buffer. Number of frames dropped can be read from register 8.130.
0	0x0	RO	**Port status vector - Bit 0**: Indicates that at least one frame was been received with CRC error. Number of frames dropped can be read from register 8.119.
NOTE: Each status bit is asserted when the corresponding event has happened and the corresponding mask bit is '1'.			

8.7. Ethernet Port

There is information about port configuration and the Transparent Clock Peer-to-peer mechanism. Each Ethernet port can be configured independently. A list of the registers is shown in Tables 8.104 and 8.105.

Table 8.104.: **Ethernet Port registers (Table 1)**

Address (Hex)	Description
0x00	Module Version
0x04	P2P control
0x08	Latency TX/RX 10Mbps
0x0C	Latency TX/RX 100Mbps
0x10	Latency TX/RX 1000Mbps
0x14	Calculated Path Delay
0x18	Half Duplex
0x1C	Measured PHY Speed
0x20	Received Frames
0x24	Transmitted Frames
0x28	Frame Errors (CRC)
0x2C-0x38	Reserved
0x3C	Initialize Statistic Counters
0x40	Enable Statistic Counters
0x44	RX Dropped Overflowed Frames (*)
0x48	RX Unicast Frames (*)
0x4C	RX Multicast Frames (*)
0x50	RX Broadcast Frames (*)
0x54	RX VLAN Tagged Frames (*)
0x58	RX PTP Frames (*)
0x5C	RX Overlength Frames (*)
0x60	RX Underlength Frames (*)
0x64	Received Data Bytes
0x68	Enable/Disable port
0x6C	VLAN Statistic Configuration (*)
NOTE: (*) These registers are only implemented when Complete Statistic Counters is selected in the GUI. Otherwise, reading these registers will have no effect.	

Table 8.105.: **Ethernet Port registers (Table 2)**

Address (Hex)	Description
0x70	TX Dropped Overflowed Frames (*)
0x74	TX Unicast Frames (*)
0x78	TX Multicast Frames (*)
0x7C	TX Broadcast Frames (*)
0x80	TX VLAN Tagged Frames (*)
0x84	TX PTP Frames (*)
0x88	Transmitted Data Bytes
0x8C - 0xBC	Reserved
NOTE: (*) These registers are only implemented when Complete Statistic Counters is selected in the GUI. Otherwise, reading these registers will have no effect.	

The contents of each configuration register are shown in Tables 8.106 to 8.136.

Table 8.106.: **Module Version [0x00]**

Bits	Default Value	Type	Description
31:0	0x19090000	RO	**Module Version**: Version number for internal use.

Table 8.107.: **P2P Control [0x04]**

Bits	Default Value	Type	Description
31	0x0	R/W	**P2P enable**: When this bit is set to 1 and the *IEEE1588 Support* is set to Transparent Clock P2P, the Peer-to-peer delay mechanism is enabled.
30	0x0	R/W	**Power Profile enable**: When this bit is set to 1, the VLAN tag is added to the Peer to-peer frames.
29:26	0x1	R/W	**P2P request period**: It specifies the number of *Pdelay_req* frames transmitted per second. The possible values are 1, 2, 4 or 8.
25:24	0x0	RO	Reserved
23:16	0x0	R/W	**SourcePortID**: It specifies the source port identification used in the Peer-to-peer frames
15:0	0x8000	R/W	**PP VLAN ID and priority**: When the Power Profile bit is set to 1, it specifies the VLAN ID field and priority of the VLAN tag added to the Peer-to-peer frames.

Table 8.108.: **Latency TX/RX 10Mbps [0x08]**

Bits	Default Value	Type	Description
31:16	0x0032	R/W	**TX latency**: ns value relating to the transmission latency inserted by PHY (10Mbps speed) which is used in the residence time calculation.
15:0	0x00F0	R/W	**RX latency**: ns value relating to the reception latency inserted by PHY (10Mbps speed) which is used in the residence time calculation.

Table 8.109.: **Latency TX/RX 100Mbps [0x0C]**

Bits	Default Value	Type	Description
31:16	0x0032	R/W	**TX latency**: ns value relating to the transmission latency inserted by PHY (100Mbps speed) which is used in the residence time calculation.
15:0	0x00F0	R/W	**RX latency**: ns value relating to the reception latency inserted by PHY (100Mbps speed) which is used in the residence time calculation.

Table 8.110.: **Latency TX/RX 1000Mbps [0x10]**

Bits	Default Value	Type	Description
31:16	0x0032	R/W	**TX latency**: ns value relating to the transmission latency inserted by PHY (1000Mbps speed) which is used in the residence time calculation.
15:0	0x00F0	R/W	**RX latency**: ns value relating to the reception latency inserted by PHY (1000Mbps speed) which is used in the residence time calculation.

Table 8.111.: **Path Delay [0x14]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Path delay**: ns value which shows the calculated Path delay of a port using the Peer-to-peer mechanism.

Table 8.112.: **Half Duplex [0x18]**

Bits	Default Value	Type	Description
31:1	0x0	RO	Reserved.
0	0x0	R/W	**Half duplex enable**: If 1 and *Half Duplex* is set to true, the IP operates in half-duplex mode. If 0, the IP operates in full-duplex mode.

Table 8.113.: **Measured PHY Speed [0x1C]**

Bits	Default Value	Type	Description
31:2	0x0	RO	Reserved.
1:0	0x0	RO	It shows the speed (negotiated by the PHY) measured by the IP core. "11" - GMII (base1000) "10" - MII (base100) "01" - MII (base10)

Table 8.114.: **Received Frames [0x20]**

Bits	Default Value	Type	Description
31:0	0x0	RO	It shows the number of received frames. In this number, the erroneous frames are also considered.

Table 8.115.: **Transmitted Frames [0x24]**

Bits	Default Value	Type	Description
31:0	0x0	RO	It shows the number of transmitted frames.

Table 8.116.: **Frame Errors (CRC) [0x28]**

Bits	Default Value	Type	Description
31:0	0x0	RO	It shows the number of frames received with erroneous CRC.

Table 8.117.: **Initialize Statistic Counters [0x3C]**

Bits	Default Value	Type	Description
31:21	0x0	NA	Reserved.
20	0x0	WO	**Initialize Transmitted Data Bytes Counter**: It initializes the counter of transmitted data bytes (CRC bytes are also considered).
19	0x0	WO	**Initialize TX PTP Frames Counter**: It initializes the counter of transmitted PTP frames.
18	0x0	WO	**Initialize VLAN Tagged Frames Counter**: It initializes the counter of received VLAN tagged frames.
17	0x0	WO	**Initialize TX Broadcast Frames Counter**: It initializes the counter of transmitted broadcast frames.
16	0x0	WO	**Initialize TX Multicast Frames Counter**: It initializes the counter of transmitted multicast frames.
15	0x0	WO	**Initialize TX Unicast Frames Counter**: It initializes the counter of transmitted unicast frames.
14	0x0	WO	**Initialize TX Dropped Overflowed Frames Counter**: It initializes the counter of dropped frames (in the transmission path) due to overflow situation.
13	0x0	WO	**Initialize Received Data Bytes Counter**: It initializes the counter of received data bytes (CRC bytes are also considered).
12	0x0	WO	**Initialize RX Underlength Frames Counter**: It initializes the counter of received undersized frames.
11	0x0	WO	**Initialize RX Overlength Frames Counter**: It initializes the counter of received oversized frames.
10	0x0	WO	**Initialize RX PTP Frames Counter**: It initializes the counter of received PTP frames.
9	0x0	WO	**Initialize VLAN Tagged Frames Counter**: It initializes the counter of received VLAN tagged frames.
8	0x0	WO	**Initialize RX Broadcast Frames Counter**: It initializes the counter of received broadcast frames.
7	0x0	WO	**Initialize RX Multicast Frames Counter**: It initializes the counter of received multicast frames.
6	0x0	WO	**Initialize RX Unicast Frames Counter**: It initializes the counter of received unicast frames.
5	0x0	WO	**Initialize RX Dropped Overflowed Frames Counter**: It initializes the counter of dropped frames (in the reception path) due to overflow situation.
4	0x0	NA	Reserved.
3	0x0	WO	**Initialize Frame Errors (CRC) Counter**: It initializes the counter of erroneous frames received .
2	0x0	WO	**Initialize Transmitted Frames Counter**: It initializes the counter of transmitted frames.
1	0x0	WO	**Initialize Received Frames Counter**: It initializes the counter of received frames.
0	0x0	WO	**Initialize All Statistics Counters**: It initializes all Statistic Counters.

Table 8.118.: **Enable Statistic Counters [0x40]**

Bits	Default Value	Type	Description
31:1	0x0	NA	Reserved.
0	0x1	R/W	**Enable All Statistic Counters**: It enables all Statistic Counters.

Table 8.119.: **RX Dropped Overflowed Frames [0x44]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Dropped Overflowed Frames**: It shows the number of dropped frames (in the reception path) due to overflow situation.

Table 8.120.: **RX Unicast Frames [0x48]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Unicast Frames**: It shows the number of received frames which contain a unicast address in the destination address field.

Table 8.121.: **RX Multicast Frames [0x4C]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Multicast Frames**:It shows the number of received frames which contain a multicast address in the destination address field.

Table 8.122.: **RX Broadcast Frames [0x50]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Broadcast Frames**: It shows the number of received frames which contain a broadcast address in the destination address field.

Table 8.123.: **RX VLAN Tagged Frames [0x54]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**VLAN Tagged Frames**: It shows the number of frames received with a VLAN identifier in the length/type field and depending on the 8.129 register configuration.

Table 8.124.: **RX PTP Frames [0x58]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**PTP Frames**: It shows the number of frames received with a PTP identifier in the length/type field.

Table 8.125.: **RX Overlength Frames [0x5C]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Overlength Frames**: It shows the number of received frames that exceed the maximum frame length (without preamble and FCS fields). It only has sense when JUMBO frame support is disabled.

Table 8.126.: **RX Underlength Frames [0x60]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Underlength Frames**: It shows the number of received frames that do not reach the minimum frame length (without preamble and FCS fields).

Table 8.127.: **Received Data Bytes [0x64]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Received Data Bytes**: It shows the number of data bytes received (FCS bytes are included but not the preamble bytes).

Table 8.128.: **Enable/Disable port [0x68]**

Bits	Default Value	Type	Description
31:1	0x0	NA	Reserved.
0	0x1	R/W	**Enable/Disable port**: It enables ('1') or disables ('0') the port. It is enabled by default.

Table 8.129.: **VLAN Statistic Configuration [0x6C]**

Bits	Default Value	Type	Description
31:13	0x0	NA	Reserved.
12	0x0	R/W	**VLAN Statistic Configuration**: It defines if all VLAN tagged frames will be counted or only those ones that matches with the set VLAN ID. If it is set to '0', all VLAN tagged frames are counted. If '1', only those filtered frames.
11:0	0x0	R/W	**VLAN ID**: It defines the VLAN ID used to filter VLAN tagged frames for statistic purposes.

Table 8.130.: **TX Dropped Overflowed Frames [0x70]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Dropped Overflowed Frames**: It shows the number of dropped frames (in the transmission path) due to overflow situation.

Table 8.131.: **TX Unicast Frames [0x74]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Unicast Frames**: It shows the number of transmitted frames which contain a unicast address in the destination address field.

Table 8.132.: **TX Multicast Frames [0x78]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Multicast Frames**: It shows the number of transmitted frames which contain a multicast address in the destination address field.

Table 8.133.: **TX Broadcast Frames [0x7C]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Broadcast Frames**: It shows the number of transmitted frames which contain a broadcast address in the destination address field.

Table 8.134.: **TX VLAN Tagged Frames [0x80]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**VLAN Tagged Frames**: It shows the number of frames transmitted with a VLAN identifier in the length/type field and depending on the 8.129 register configuration.

Table 8.135.: **TX PTP Frames [0x84]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**PTP Frames**: It shows the number of frames transmitted with a PTP identifier in the length/type field.

Table 8.136.: **Transmitted Data Bytes [0x88]**

Bits	Default Value	Type	Description
31:0	0x0	RO	**Transmitted Data Bytes**: It shows the number of data bytes transmitted (FCS bytes are included but not the preamble bytes).

9. Constraining the Core

This chapter contains information about constraining the core in the Vivado Design Suite.

These constraints should be placed in an XDC file at the top level of the design. The example of the constraint text shown in the following paragraphs is based on the port names of the **ME**S IP core. If these ports are mapped to FPGA pin names that are different, the FPGA pin names should be submitted for the port names in the following example.

Note that these constraints are hardware dependant and should be adapted to the customer needs.

9.1. RGMII Interface Timing Constraints

```
### NOTE ###
# "clk_out1_design_1_clk_wiz_0_0" is the clock used by the IP core

##### RGMII receive clocks #####
create_clock -period 8.000 -name port_0_rgmii_rxc [get_ports port_0_rgmii_rxc]

##### Virtual clocks for Rx #####
create_clock -period 8.000 -name rx_virt_clk

##### Virtual clocks for Tx ##### # TX clock is regenerated using the RX clock
create_generated_clock -name port_0_rgmii_txc -source [get_pins -of [get_cells -hier -filter name =
*inst_port_interface_external_0*gmii_tx_clk_ddr_iob_ddr] -filter name =  *C] -divide_by 1 -add -master_clock
port_0_rgmii_rxc [get_ports port_0_rgmii_txc]

#-----------------------------------------------------------
# RGMII
# The data window associated with the rising edge starts 1.5ns before the rising edge and ends 1.2ns
after the rising edge.
# The data window is captured with the rising edge of the clock. To do this, it is need that
# - set_multicycle_path 0
# - set_multicycle_path -1 -hold
# - set_false_path between different combinations of the external and internal rising and falling clock
```

edges for setup and hold checks
 # ————————————————————————————————

RX input delay
set_input_delay -clock [get_clocks rx_virt_clk] -min -2.800 [get_ports port_0_rgmii_rxd[*] port_0_rgmii_rx_ctl]
set_input_delay -clock [get_clocks rx_virt_clk] -max -1.500 [get_ports port_0_rgmii_rxd[*] port_0_rgmii_rx_ctl]
set_input_delay -clock [get_clocks rx_virt_clk] -clock_fall -min -add_delay -2.800 [get_ports port_0_rgmii_rxd[*] port_0_rgmii_rx_ctl]
set_input_delay -clock [get_clocks rx_virt_clk] -clock_fall -max -add_delay -1.500 [get_ports port_0_rgmii_rxd[*] port_0_rgmii_rx_ctl]

Set_false_path between different combinations of the external and internal rising and falling clock edges for setup and hold checks
set_false_path -setup -rise_from [get_clocks rx_virt_clk] -fall_to [get_clocks port_?_rgmii_rxc]
set_false_path -setup -fall_from [get_clocks rx_virt_clk] -rise_to [get_clocks port_?_rgmii_rxc]
set_false_path -hold -rise_from [get_clocks rx_virt_clk] -rise_to [get_clocks port_?_rgmii_rxc]
set_false_path -hold -fall_from [get_clocks rx_virt_clk] -fall_to [get_clocks port_?_rgmii_rxc]

The data window is captured with the rising edge of the clock, so pulls the capture edge back by one full clock period in order to correct capture edges
set_multicycle_path -setup -from [get_clocks rx_virt_clk] -to port_?_rgmii_rxc 0
set_multicycle_path -hold -from [get_clocks rx_virt_clk] -to port_?_rgmii_rxc -1

Output delays
set_output_delay -clock [get_clocks port_0_rgmii_txc] -clock_fall -min -add_delay -2.600 [get_ports port_0_rgmii_txd[*] port_0_rgmii_tx_ctl]
set_output_delay -clock [get_clocks port_0_rgmii_txc] -clock_fall -max -add_delay -1.000 [get_ports port_0_rgmii_txd[*] port_0_rgmii_tx_ctl]
set_output_delay -clock [get_clocks port_0_rgmii_txc] -min -add_delay -2.600 [get_ports port_0_rgmii_txd[*] port_0_rgmii_tx_ctl]
set_output_delay -clock [get_clocks port_0_rgmii_txc] -max -add_delay -1.000 [get_ports port_0_rgmii_txd[*] port_0_rgmii_tx_ctl]

Added CDCs constraints
set_max_delay -datapath_only -from [get_clocks clk_out1_design_1_clk_wiz_0_0] -to [get_clocks port_?_rgmii_rxc] 4.000
set_max_delay -datapath_only -from [get_clocks port_?_rgmii_rxc] -to [get_clocks clk_out1_design_1_clk_wiz_0_0] 4.000

Mutually exclusive clocks
set_clock_groups -asynchronous -group [get_clocks port_0_rgmii_rxc] -group [get_clocks port_0_rgmii_txc]

IDELAYS
*set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =~ *delay_gmii_rx_dv]*

```
set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =  *delay_gmii_rxd*]
set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =  *util_idelay_ctrl_0]
set_property IDELAY_VALUE 12 [get_cells -hier -filter name =  *delay_gmii_rx_dv]
set_property IDELAY_VALUE 12 [get_cells -hier -filter name =  *delay_gmii_rxd*]
```

9.2. GMII Interface Timing Constraints

```
### NOTE ###
# "clk_out1_design_1_clk_wiz_0_0" is the clock used by the IP core

##### GMII receive/transmit clocks #####
create_clock -period 40.000 -name port_0_gmii_tx_clk [get_ports port_0_gmii_tx_clk]
create_clock -period 8.000 -name port_0_gmii_rx_clk [get_ports port_0_gmii_rx_clk]

##### Virtual clocks for Rx #####
create_clock -period 8.000 -name rx_virt_clk

##### Virtual clocks for Tx #####
# TX clock is regenerated using the RX clock

create_generated_clock -name port_0_gmii_gtx_clk -source [get_pins -of [get_cells -hier -filter name =
*inst_port_interface_external_0*gmii_tx_clk_ddr_iob_sdr] -filter name =  *C] -divide_by 1 -invert -add -
master_clock port_0_gmii_rx_clk [get_ports port_0_gmii_gtx_clk]

##### RX input delay #####
set_input_delay -clock [get_clocks rx_virt_clk] -min 3.800 [get_ports port_0_gmii_rxd[*] port_0_gmii_rx_dv]
set_input_delay -clock [get_clocks rx_virt_clk] -max 3.800 [get_ports port_0_gmii_rxd[*] port_0_gmii_rx_dv]

##### Output delays #####
set_output_delay -clock [get_clocks port_0_gmii_gtx_clk] -min -add_delay 2.600 [get_ports port_0_gmii_txd[*]
port_0_gmii_tx_en]
set_output_delay -clock [get_clocks port_0_gmii_gtx_clk] -max -add_delay 1.000 [get_ports port_0_gmii_txd[*]
port_0_gmii_tx_en]

##### Added CDCs constraints #####
set_max_delay -datapath_only -from [get_clocks clk_out1_design_1_clk_wiz_0_0] -to [get_clocks port_?_gmii_rx_clk]
8.000
set_max_delay -datapath_only -from [get_clocks port_?_gmii_rx_clk] -to [get_clocks clk_out1_design_1_clk_wiz_0_0]
8.000

##### Mutually exclusive clocks #####
set_clock_groups -asynchronous -group [get_clocks port_0_gmii_rx_clk] -group [get_clocks port_0_gmii_gtx_clk]
set_clock_groups -asynchronous -group [get_clocks port_?_gmii_rx_clk] -group [get_clocks port_?_gmii_tx_clk]
set_clock_groups -asynchronous -group [get_clocks clk_out1_design_1_clk_wiz_0_0] -group [get_clocks port_?_gmii_tx_clk]

##### IDELAYS #####
set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =  *delay_gmii_rx_dv]
set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =  *delay_gmii_rxd*]
set_property IODELAY_GROUP gpr1 [get_cells -hier -filter name =  *util_idelay_ctrl_0]
set_property IDELAY_VALUE 12 [get_cells -hier -filter name =  *delay_gmii_rx_dv]
set_property IDELAY_VALUE 12 [get_cells -hier -filter name =  *delay_gmii_rxd*]
```

9.3. MII Interface Timing Constraints

```
### NOTE ###
# "clk_out1_design_1_clk_wiz_0_0" is the clock used by the IP core

##### MII receive/transmit clocks #####
create_clock -period 40.000 -name port_0_mii_tx_clk [get_ports port_0_mii_tx_clk]
create_clock -period 40.000 -name port_0_mii_rx_clk [get_ports port_0_mii_rx_clk]

##### Virtual clocks for Rx #####
create_clock -period 40.000 -name rx_virt_clk

##### virtual clocks for Tx #####
create_clock -period 40.000 -name tx_virt_clk

##### RX input delay #####
set_input_delay -clock [get_clocks rx_virt_clk] -min 5.800 [get_ports port_0_mii_rxd[*] port_0_mii_rx_dv]
set_input_delay -clock [get_clocks rx_virt_clk] -max 5.500 [get_ports port_0_mii_rxd[*] port_0_mii_rx_dv]

##### Output delays #####
set_output_delay -clock [get_clocks tx_virt_clk] -min -add_delay 2.600 [get_ports port_0_mii_txd[*] port_0_mii_tx_en]
set_output_delay -clock [get_clocks tx_virt_clk] -max -add_delay 1.000 [get_ports port_0_mii_txd[*] port_0_mii_tx_en]

##### Added CDCs constraints #####
set_max_delay -datapath_only -from [get_clocks clk_out1_design_1_clk_wiz_0_0] -to [get_clocks port_?_mii_rx_clk]
8.000
set_max_delay -datapath_only -from [get_clocks port_?_mii_rx_clk] -to [get_clocks clk_out1_design_1_clk_wiz_0_0]
8.000
set_max_delay -datapath_only -from [get_clocks clk_out1_design_1_clk_wiz_0_0] -to [get_clocks port_?_mii_tx_clk]
8.000
set_max_delay -datapath_only -from [get_clocks port_?_mii_tx_clk] -to [get_clocks clk_out1_design_1_clk_wiz_0_0]
8.000

##### Mutually exclusive clocks #####
set_clock_groups -asynchronous -group [get_clocks port_0_mii_rx_clk] -group [get_clocks port_0_mii_tx_clk]
```

10. Resources

This chapter shows different implementations of the **ME**S IP core to provide some references of the resource usage (Table 10.1). The targeted device in these implementations is the Zynq XC7Z020-1CLG484C applying the following configurations parameters:

- Four Tri-speed port **ME**S with complete statistics.
- Four Tri-speed port **ME**S with four priority queues and complete statistics.
- Four Tri-speed port **ME**S with IEEE 1588-2008 V2 Transparent Clock Peer-to-Peer and complete statistics.

Table 10.1.: **Resource usage.**

Configuration	Slice	LUT			BRAM	DSP
		Logic	Memory	FF		
MES : - 4 ports	3520 (26.47%)	9394 (17.66%)	321 (1.84%)	11956 (22.47%)	47 (33.57%)	0 (0%)
MES : - 4 ports - 4 priority queues	4465 (33.57%)	12624 (23.73%)	465 (2.67%)	15219 (28.61%)	57 (40.71%)	0 (0%)
MES : - 4 ports - TC-P2P	5316 (39.97%)	13947 (26.22%)	381 (2.19%)	18559 (34.89%)	47 (33.57%)	0 (0%)

NOTE: The resource usage could have changed since the edition of this document.

11. Performance

This chapter shows some performance results obtained from different tests carried out over **ME***S* IP core.

11.1. Measured latencies

This section shows latency measurements carried out in simulation (Table 11.1) for different frame lengths and PHY speeds. This latency value is the time lapsed between the first byte of the received frame and the first byte of the transmitted frame (considering only the IP logic).

Table 11.1.: **ME***S* **IP core latencies for different frame lengths.**

Frame length (without CRC)	**Latency**		
	1000 Mbps	100 Mbps	10 Mbps
64 bytes	0.992 μs	6.88 μs	66 μs
256 bytes	2.528 μs	22.24 μs	219.6 μs
512 bytes	4.576 μs	42.72 μs	424.4 μs
1024 bytes	8.672 μs	83.68 μs	834 μs
1518 bytes	12.632 μs	123.2 μs	1229.2 μs

NOTE: these results are estimated values which can vary from a real application.

A. Appendix A

This appendix contains information about how to manage MESS IP Core for its different features. The descriptions are accompanied with examples.

A.1. Configuration of the VLAN

The MESS IP Core includes support for port based VLAN switching. In Port-based Virtual LAN two basic roles for ports are configured. Depending on this configuration, the process of being added into a VLAN and the way of forwarding packets are different:

- *Access port*: a port configured as untagged port can belong to only one VLAN at a time and is designed to provide connectivity for an end device. The frames received on these ports are known as untagged frames due to the fact that the frames received on a port do not contain any information that indicates VLAN membership. This information is determined solely by ports PVID (port VLAN identifier). By default, these ports are configured with VLAN ID 1.
- *Trunk port*: it interconnects switches to allow devices attached to a particular switch to communicate with devices attached to another switch. Trunking allows switches to transmit traffic from multiple VLANs configured locally across the trunk (over a single connection), thus allowing a VLAN to be distributed over multiple switches.

In order to apply such operation modes, the following MESS IP core registers need to be configured appropriately (Table 8.9):

- *Forwarding Behaviour Configuration* (0x04)
- *Native VLAN* (0x0C)
- *VLAN ID mask configuration* (0x10)
- *VLAN ID mask* (0xE0)

The VLAN configuration consists basically on three steps managing these MESS Control Registers. The correct order to configure VLANs will be:

- **1. step:** *Native VLAN* (0x0C)
- **2. step:** *VLAN ID mask* (0xE0)
- **3. step:** *VLAN ID mask configuration* (0x10)
- **4. step:** *Forwarding Behaviour Configuration* (0x04)

A.1.1. Setting VLAN Configuration

Following it is described the steps to configure the VLANs on the **MES** IP core by means of an example. It is considered a three port **MES** with the following VLAN configuration:

- Port mode:
  - Port 0 → **Access port**
  - Port 1 → **Access port**
  - Port 2 → **Trunk port**
- Native VLAN IDs:
  - Port 0 → **VLAN ID 100**
  - Port 1 → **VLAN ID 200**
  - Port 2 → **VLAN ID 300**
- Allowed VLAN IDs:
  - Port 0 (access port) → **VLAN ID 100**
  - Port 1 (access port) → **VLAN ID 200**
  - Port 2 (trunk port) → **VLAN ID 100, 200 and 300**

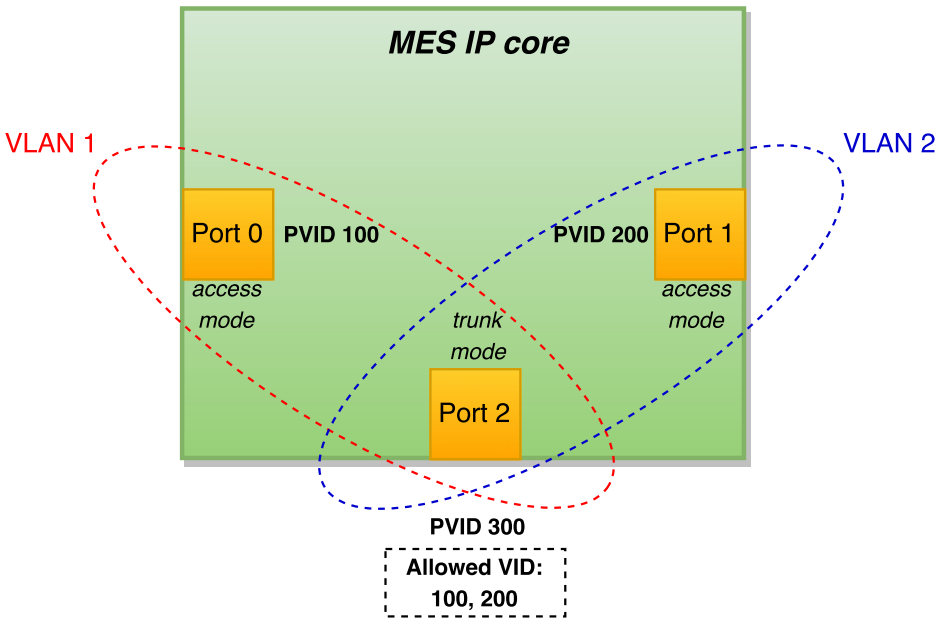


Figure A.1.: **VLAN configuration example.**

The steps that should be done to configure the above mentioned VLAN configuration are:

- 1. step: Configure the Native VLAN ID for each port. The register *Native VLAN* (Table 8.14) is used for this purpose. In the case of the example shown in the Figure A.1, **0x000000C9** should be written in this register to configure VLAN ID 100 for Port 0:

Table A.1.: ***Native VLAN*** **register write**

Bits	Value
31:27	Port zero: **"00000"**
26:17	Not used.
16:14	VLAN PCP priority bits: **"000"** (Priority zero)
13	VLAN CFI: **'0'**
12:1	VLAN ID 100: **x"064"**
0	Write process: **'1'**

  The same step 1 should be follow to configure each Native VLAN ID for each port. In the example shown in the figure A.1, the values that have to be set are:
  - VLAN ID 100 for Port 0:
    Native VLAN → **0x000000C9**
  - VLAN ID 200 for Port 1:
    Native VLAN → **0x08000191**
  - VLAN ID 300 for Port 2:
    Native VLAN → **0x10000259**

- 2. step: Configure the VLAN ID masks. Specify which ports belong to the configured VLAN IDs. The registers *VLAN ID mask configuration* (Table 8.15) and *VLAN ID mask* (Table 8.63) are used for this purpose. In the case of the example shown in the Figure A.1, this is what should be written in these registers in order to configure the VLAN mask associated to the VLAN ID 100 (Ports 0 and 2 belong to VLAN ID 100):
  - 2.1. step: Set that ports 0 and 2 belong to VLAN ID 100 writing **0x00000005** in the register *VLAN ID mask* (Table 8.63).

Table A.2.: ***VLAN ID mask*** **register write**

Bits	Value
31:3	Not used in this example
2:0	Port 0 and 2 belong to VLAN ID 100: **"101"**

  - 2.2. step: Apply that ports 0 and 2 belong to VLAN ID 100 writing **0x06400001** in the register *VLAN ID mask configuration* (Table 8.15).

Table A.3.: ***VLAN ID mask configuration*** **register write**

Bits	Value
31:20	VLAN ID 100: **x"064"**
19:2	Not used.
1	Not a "clear" command: **'0'**
0	Write operation: **'1'**

The same steps 2.1 and 2.2 should be follow to configure each VLAN mask associated to the configured VLAN IDs. For the example shown in the figure A.1, the values that have to be set are:

- Port 0 and 2 are in the VLAN ID 100:
  VLAN ID mask → **0x00000005**
  VLAN ID mask configuration → **0x06400001**
- Port 1 and 2 are in the VLAN ID 200:
  VLAN ID mask → **0x00000006**
  VLAN ID mask configuration → **0x0C800001**
- Port 2 is in the VLAN ID 300:
  VLAN ID mask → **0x00000004**
  VLAN ID mask configuration → **0x12C00001**

- 3. step: Enable port based VLAN functionality and configure ports 0 and 1 as *access ports* and port 2 as *trunk port* writing **0x80000004** in the register *Forwarding Behaviour Configuration* (Table 8.13).

Table A.4.: ***Forwarding Behaviour Configuration*** **register write**

Bits	Value
31	Enable Port based Virtual VLAN: **'1'**
30	Disable Switching port mask: **'0'**
29:3	Not used.
2:0	Port mode selector: **"100"** (Ports 0 and 1 in access mode - '0'; Port 2 in trunk mode - '1')

A.1.2. Reading VLAN Configuration

Using same registers, it is possible to read back the set VLAN configuration. For example, the operations that should be done to read the configured native VLAN ID in the port 0 are:

- 1. step: Write in the *Native VLAN* register (Table 8.14) the value **0x00000000** (Table A.5) meaning that a read operation over the port 0 is going to be done in order to obtain its native VLAN ID.

Table A.5.: ***Native VLAN* register write - Read command**

Bits	Value
31:27	Port zero: **"00000"**
26:17	Not used.
16:14	Not specified
13	Not specified
12:1	VLAN ID: Not specified
0	Read process: **'0'**

- 2. step: Read the *Native VLAN* register in order to obtain the configured native VLAN ID of the consulted port. In this case, the value **0x064** should be read in the *VLAN ID* field (bits 12:1).

A.1.3. Forwarding frames based on VLAN Configuration

Depending on the VLAN and port mode configuration that has been set, the switch will forward frames in different way. Using the VLAN configuration example shown in the Figure A.1:

- Port 0:
  1. It receives an untagged frame: it is tagged with VLAN ID 100 and is only transmitted through port 2 because the VLAN ID 100 is in its allowed VLANs. The inserted VLAN tag is not removed when it is transmitted through port 2.
  2. It receives a tagged frame that does not match with the VLAN ID 100: the frame is removed.
  3. It receives a tagged frame that matches with the VLAN ID 100: the frame is tagged again (doubly tagged) and it is only transmitted through port 2 because the VLAN ID 100 is in its allowed VLANs. The inserted VLAN tag is not removed when it is transmitted through port 2, so it will go doubly tagged.
- Port 1:
  1. It receives an untagged frame: it is tagged with VLAN ID 200 and is only transmitted through port 2 as the VLAN ID 200 is in its allowed VLANs. The inserted VLAN tag is not removed when it is transmitted through port 2.
  2. It receives a tagged frame that does not match with the VLAN ID 200: the frame is removed.
  3. It receives a tagged frame that matches with the VLAN ID 200: the frame is tagged again (double tagged) and it is only transmitted through port 2

because the VLAN ID 200 is in its allowed VLANs. The inserted VLAN tag is not removed when it is transmitted through port 2, so it will go double tagged.

- Port 2:
  1. It receives an untagged frame: it is tagged with VLAN ID 300 and is transmitted through the other ports. As it is originally an untagged frame and there is not other port in that VLAN, it is transmitted to all other ports depending on the MAC table. The inserted VLAN tag is removed when it is transmitted through port 0 and 1.
  2. It receives a tagged frame that does not match with the VLAN ID 300 nor with its allowed VLANs: the frame is removed.
  3. It receives a tagged frame that matches with the VLAN ID 300: the frame is not tagged again and is transmitted through the other ports. This tagged frame is allowed in the reception port but there is not other port in that VLAN. Thus, it is transmitted to all other ports depending on the MAC table. The inserted VLAN tag is removed when it is transmitted through port 0 and 1.
  4. It receives a tagged frame that matches with one of the allowed VLANs: the frame is not tagged again and is transmitted through the corresponding port (depending on the VLAN ID). The inserted VLAN tag is removed when it is transmitted through the corresponding port.

A.2. Configuration of Static MAC addresses

Apart from the dynamic MAC address learning, it is possible to configure and handle Static MAC addresses in the **ME***S* IP Core. Not only unicast MAC addresses can be configured, but also multicast MAC addresses. In the case of multicast addresses, more than one source port can be configured for each multicast address.

There are two type of operations that can be done:

- Configure a new Static MAC address entry
- Clear an existing Static MAC address entry

It is not possible to read back the configured Static MAC address. For this purpose, *MAC address query* functionality (A.3) has to be used. This mechanism provides the list of MAC Addresses that are contained in its Dynamic and Static MAC Tables, classified by the port number.

In order to use Static MAC address feature, the following **ME***S* IP core registers need to be configured appropriately (Tables 8.9 and 8.10):

- *Static MAC address LO* (0x78)
- *Static MAC address HI and control* (0x7C)
- *Static MAC origin ports* (0xE8)

The Static MAC address configuration consists basically on three steps managing these **ME***S* Control Registers. The correct order to configure Static MAC addresses will be:

- **1. step:** *Static MAC origin ports* (0xE8)
- **2. step:** *Static MAC address LO* (0x78)
- **3. step:** *Static MAC address HI and control* (0x7C)

A.2.1. Setting a Static MAC address

Following it is described the steps to configure Static MAC addresses on the **ME***S* IP core by means of an example. It is considered a three port **ME***S* in which *0x50CE50CE50CE* Static MAC address is configured in port 2:

- 1. step: Configure the origin port(s) of the Static MAC address that is going to be configured. The register *Static MAC origin ports* (Table 8.65) is used for this purpose. In the case of this example, **0x00000004** should be written in this register to configure port 2 as the origin port of the Static MAC address:

Table A.6.: ***Static MAC origin ports*** **register write**

Bits	Value
31:3	Not used in this example.
2:0	Origin port(s): **"100"** (Port 2)

- 2. step: Set the 32 lowest bits of the Static MAC address in the *Static MAC address LO* register (Table 8.38) writing **0x50CE50CE**.

Table A.7.: ***Static MAC address LO*** **register write**

Bits	Value
31:0	32 lowest bits of the MAC address: **0x50CE50CE**

- 3. step: Set the 16 highest bits of the Static MAC address in the *Static MAC address HI and control* registers (Table 8.39) and command a write operation. In the case of this example, **0x000150CE** should be written:

Table A.8.: ***Static MAC address HI and control*** **register write**

Bits	Value
31:18	Not used.
17	Not a clear process: **'0'**
16	Write process: **'1'**
15:0	16 highest bits of the MAC address: **0x50CE**

A.2.2. Removing a Static MAC address

Similar to the way in which a Static MAC address is set, it is possible to clear a specific static MAC address. These are the operations that should be done to clear the Static MAC address *0x50CE50CE50CE* from port 2:

- 1. step: Set the 32 lowest bits of the Static MAC address that is going to be cleared in the *Static MAC address LO* register (Table 8.38) writing **0x50CE50CE**.

Table A.9.: ***Static MAC address LO*** **register write**

Bits	Value
31:0	32 lowest bits of the MAC address: **0x50CE50CE**

- 2. step: Set the 16 highest bits of the Static MAC address that is going to be cleared in the *Static MAC address HI and control* registers (Table 8.39) and command a clear operation. In the case of this example, **0x000250CE** should be written:

Table A.10.: ***Static MAC address HI and control*** **register write**

Bits	Value
31:18	Not used.
17	Clear process: **'1'**
16	Not a write process: **'0'**
15:0	16 highest bits of the MAC address: **0x50CE**

A.3. MAC Address Query

The **MES** IP Core supports a mechanism of providing the list of MAC Addresses that are contained in its MAC Address table, classified by the port number. These queries can be done to both Dynamic and Static MAC Tables.

There are four type of operations that can be done:

- Select between Dynamic and Static MAC Table.
- Start a new MAC address query. It will discard any MAC address query that would be under process.
- Continue with the MAC address query under process.
- Cancel an existing MAC address query (before all the MAC addresses have been read).

In order to use MAC query feature, the following **MES** module registers need to be configured appropriately (Table 8.9):

- *MAC address query control* (0x38)
- *MAC address query HI* (0x3C)
- *MAC address query LO* (0x40)

The MAC Address Query on a specific port consists basically on three steps managing these **MES** Control Registers. The correct order to retrieve Static MAC addresses will be:

- **1. step:** *MAC address query control* (0x38)
- **2. step:** *MAC address query HI* (0x3C)
- **3. step:** *MAC address query LO* (0x40)

A.3.1. Retrieving the list of MAC addresses

Following it is described the steps to retrieve the list of MAC Addresses that are contained in the MAC Address table of the **MES** IP core by means of an example. It is considered a three port **MES** in which there are the following MAC addresses in its Dynamic MAC Table for port 2 (Figure A.2):

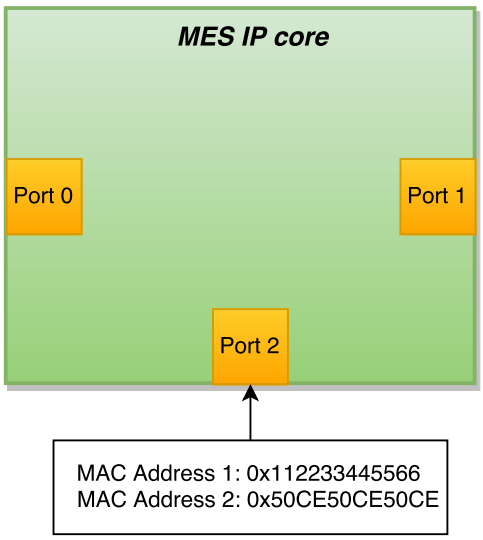


Figure A.2.: **MAC Address query example.**

The steps that should be done to perform a MAC address query are:

- 1. step: Start a new MAC address query for a specific port. The register *MAC address query control* (Table 8.24) is used for this purpose. In the case of the example shown in the Figure A.2, **0x00000021** should be written in this register to retrieve the MAC addresses located in the Port 2:

Table A.11.: ***MAC address query control*** **register write**

Bits	Value
31:9	Not used
8:4	Port under consult: **"00010"** (Port 2)
3	Select Dynamic MAC Table: **'0'**
2	Cancel MAC query: **'0'**
1	Continue MAC query: **'0'**
0	Start MAC query: **'1'**

- 2. step: Read the *MAC address query HI* register (Table 8.25) in order to obtain the upper 16 bits of the MAC address associated to port 2. In this case, the value **0x1122** should be read in the *Upper 16 bits of the MAC address* field (bits 31:16) for the first MAC address.

  There are also some status bits on this register:

  – *Port selector*: This field indicates which port is being queried.
  – *MAC address consultation valid*: When this bit is set to '1' the value of *MAC address query HI* (Table 8.25) and *MAC address query LO* (Table 8.26) registers is valid.

– *MAC address consultation finished*: This bit is set to '1' when there are no more MAC addresses associated to the port where the query is being performed.

Table A.12.: ***MAC address query HI*** **register read**

Bits	Value
31:16	Upper 16 bits of the MAC address associated to port 2: **x"1122"**
15:7	Reserved
6:2	Port under consult: **"00010"** (Port 2)
1	MAC query valid: **'1'**
0	MAC query finished: **'0'**

- 3. step: Read the *MAC address query LO* register (Table 8.26) in order to obtain the lower 32 bits of the MAC address associated to port 2. In this case, the value **0x33445566** should be read for the first MAC address.

Table A.13.: ***MAC address query LO*** **register read**

Bits	Value
31:0	Lower 32 bits of the MAC address associated to port 2: **x"33445566"**

The value of this register is only valid when *MAC query valid* bit in Table A.12 is '1'.

- 4. step: If the *MAC query finished* bit in Table A.12 is '0', it is possible to continue retrieving more MAC addresses associated to port 2. For this purpose, the value **0x00000022** should be written in the register *MAC address query control* (Table 8.24).

Table A.14.: ***MAC address query control*** **register write**

Bits	Value
31:9	Not used
8:4	Port under consult: **"00010"** (Port 2)
3	Select Dynamic MAC Table: **'0'**
2	Cancel MAC query: **'0'**
1	Continue MAC query: **'1'**
0	Start MAC query: **'0'**

It is also possible to cancel the current consultation in any moment setting to '1' the *Cancel MAC query* bit (Table A.11).

A.4. PHY registers access through MDIO

The PHY registers are accessed through the *MDIO bridge* module which makes bridging functions between the selected configuration interface and the MDIO interfaces. The number of MDIO interfaces depends on the used IP core. In the case of the **MES** , this number can be up to n ports.

In order to command a MDIO operation over PHY registers, the *MDIO Control* register (Address 0x04, Table 8.91) is used. There are two type of operations that can be done using this register:

- *MDIO write access*: a write process over a PHY register supposes one write operation over the *MDIO Control* register. The write operation is performed in order to specify the MDIO operation type, the PHY and register addresses to be accessed and the data to be written.
- *MDIO read access*: a read process over a PHY register supposes one write and one read operation over the *MDIO Control* register. The write operation is performed first in order to specify the MDIO operation type and the PHY and register addresses to access. Then, a read operation over the *MDIO Control* register is done in order to obtain the read value.

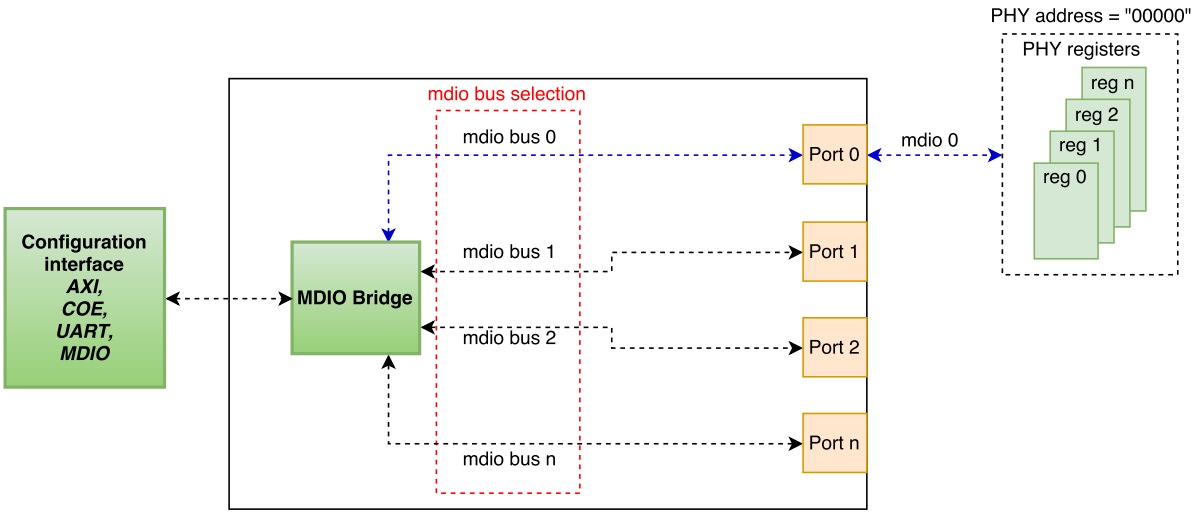


Figure A.3.: **PHY register access through MDIO.**

Following are given some examples of MDIO read and write accesses of the MARVELL *88E1512* PHY (http://www.marvell.com/documents/EoXWrBLUVWybgXVagKkF/) registers:

A.4.1. MDIO read access

As a MDIO read access example, the *PHY identifier* (page 0, register 2) is going to be read from the PHY 0 module (PHY address "00000") and through the mdio bus 0 (mdio bus selection "0000") (Figure A.3). These are the operations that should be done:

- 0. step: First of all, be sure that it is placed in the page 0 of the PHY. Use the register 22 of the PHY to specify the page that is going to be accessed. In this case, a register of the page 0 is going to be accessed so the value **0x05900000** has to be written in the *MDIO Control* register.
- 1. step: Write in the *MDIO Control* register the value **0x00B00000** (Table A.15) meaning a read operation over the mentioned PHY register.

Table A.15.: ***MDIO Control*** **register write - MDIO read operation**

Bits	Value
31:27	PHY Address: **"00000"**
26:22	Register Address: **"00010"**
21	Operation: **'1'** (Read)
20	Operation status: **'1'** (Start a new operation)
19:16	MDIO bus selection: **"0000"**
15:0	Data: **0x0000**

- 2. step: Read the *MDIO Control* register in order to obtain the value of the consulted PHY register. In this case, the value **0x0141** (PHY identifier) is read in the *Data* field (bits 15:0).

A.4.2. MDIO write access

As a MDIO write access example, the *Auto Negotiation Enable* bit of the *Cooper Control Register* (page 0, register 0 and bit 12) is going to be changed in the PHY 0 module (PHY address "00000") and through the mdio bus 0 (mdio bus selection "0000") (Figure A.3). The original value of this *Cooper Control Register* is **0x1140**. This is the operation that should be done in order to disable the Auto Negotiation:

- 0. step: First of all, be sure that it is placed in the page 0 of the PHY. Use the register 22 of the PHY to specify the page that is going to be accessed. In this case, a register of the page 0 is going to be accessed so the value **0x05900000** has to be written in the *MDIO Control* register.
- 1. step: Write in the *MDIO Control* register the value **0x00100140** (Table A.16) meaning a write operation over the mentioned PHY register and with the specified data.

Table A.16.: ***MDIO Control*** **register write - MDIO write operation**

Bits	Value
31:27	PHY Address: **"00000"**
26:22	Register Address: **"00000"**
21	Operation: **'0'** (Write)
20	Operation status: **'1'** (Start a new operation)
19:16	MDIO bus selection: **"0000"**
15:0	Data: **"0000000101000000"** → **0x0140**

- 2. step: Read back the *Cooper Control Register* (page 0, register 0) following the steps described in the *MDIO read access* paragraph in order to check that the mdio write access has been performed adequately (checking read *Data* field). It should be **0x0140**.

B. Appendix B

This appendix contains the complete reports of the *RFC 2544* tests created by *EXFO FTB-1* tester.

B.1. RFC 2544 test - 1000Mbps

FTB-860 Report

| Managed Ethernet Switch IP - RFC 2544 |

PASS

Report Header:	System-on-Chip Engineering S.L.
Report Title:	Managed Ethernet Switch IP - RFC 2544
Report Date:	11/11/2016 11:13:32
Type:	RFC 2544

Job Information

Job ID:	
Circuit ID:	
Contractor Name:	
Customer Name:	
Operator Name:	Unknown Operator
Comment:	1000Mbps Test

Table of contents

SUMMARY

Results Summary

Test Status	
Throughput	Completed 00d:00:17:43
Back-to-Back	Completed 00d:00:10:48
Frame Loss	Completed 00d:00:21:38
Latency	Completed 00d:00:10:49
Pass/Fail Verdict	PASS
Start Time	11/11/2016 09:56:43
Power Recovery	0

RFC 2544

Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)			
	Layer		
	All	**Ethernet**	**IP**
64	1000,000	761,904	547,619
128	1000,000	864,864	743,243
256	1000,000	927,536	862,318
512	1000,000	962,406	928,571
1024	1000,000	980,842	963,601
1280	1000,000	984,615	970,769
1518	1000,000	986,996	975,292

Back-to-Back (Mbit/s)			
	Layer		
	All	Ethernet	IP
64	1000,000	761,904	547,619
128	1000,000	864,864	743,243
256	1000,000	927,536	862,318
512	1000,000	962,406	928,571
1024	1000,000	980,842	963,601
1280	1000,000	984,615	970,769
1518	1000,000	986,996	975,292

	Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	0,000	10,46
128	0,000	10,99
256	0,000	12,02
512	0,000	14,07
1024	0,000	18,17
1280	0,000	20,17
1518	0,000	22,09

RX Frequency		
	Max Negative Offset	Max Positive Offset
Frequency Offset (ppm)	0,0	6,3

Alarms/Errors

Alarms/Errors List	
Active/Historical Alarms/Errors	None

Interface	
Alarms	**Seconds**
Frequency	0

Clock Sync	
Alarm	**Seconds**
LOC	0

Ethernet	
Alarms	**Seconds**
Link Down	0

Errors	Seconds	Count	Rate
Symbol	0	0	0,00E00
False Carrier	0	0	0,00E00
Idle	0	0	0,00E00
FCS	0	0	0,00E00
Jabber	0	0	0,00E00
Oversize	Disabled		
Runt	0	0	0,00E00
Undersize	0	0	0,00E00
Total		0	

IP/UDP/TCP			
Errors	**Seconds**	**Count**	**Rate**
IP Checksum	0	0	0,00E00
UDP Checksum	0	0	0,00E00
Total		0	

Setup Summary

Application Type	RFC 2544

Interface	**Port 1**
Interface/Rate	10/100/1000M Electrical
Connector	RJ45

Auto-Negotiation	Enabled
Speed	1000 Mbit/s
Duplex	Full
Flow Control	None
Local Clock	Slave

Framing: MAC/IP/UDP

IP Version	IPv4

	MAC	**IP Address**	**UDP Port**
Source	00:03:01:FE:01:19	10.10.1.25 / 255.255.0.0	49184
Destination	00:03:01:FE:01:1A	10.10.1.25	7

EtherType	0x0800
Automatic IP (DHCP)	Disabled
TTL	128
TOS	0x00
Default Gateway	Disabled

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests			
Throughput	Max. Rate (%)	Threshold (%)	Accuracy (%)
	100,0000	100,0	1,00
Back-to-Back	Burst Time (MM:SS)	Threshold (%)	Accuracy (Frames)
	00:01	100,0	1
Frame Loss	Max. Rate (%)	Threshold (%)	Granularity (%)
	100,0000	0,1	10
Latency	Max. Rate (%)		Threshold (ms)
	100,0000		125,0

TEST RESULTS

Ethernet Traffic

Frame Type	TX Count	RX Count
Multicast	0	0
Broadcast	420	0
Unicast	195744980	195744980
Non-Unicast	420	0
Total	195745400	195744980

Frame Size (Bytes)	RX Count	%
< 64	0	0
64	87797620	44
65 – 127	0	0
128 – 255	49831100	25
256 – 511	26721040	13
512 – 1023	13862770	7
1024 – 1518	17532450	8
> 1518	0	0
Total	195744980	

Flow Control Monitoring

Frame Count	
Pause Frames	0
Abort Frames	0
Frames RX	0
Frames TX	0

Pause Time (Quanta)	
Total Pause Time	0
Last Pause Time	0
Minimum Pause Time	0
Maximum Pause Time	0

RFC 2544

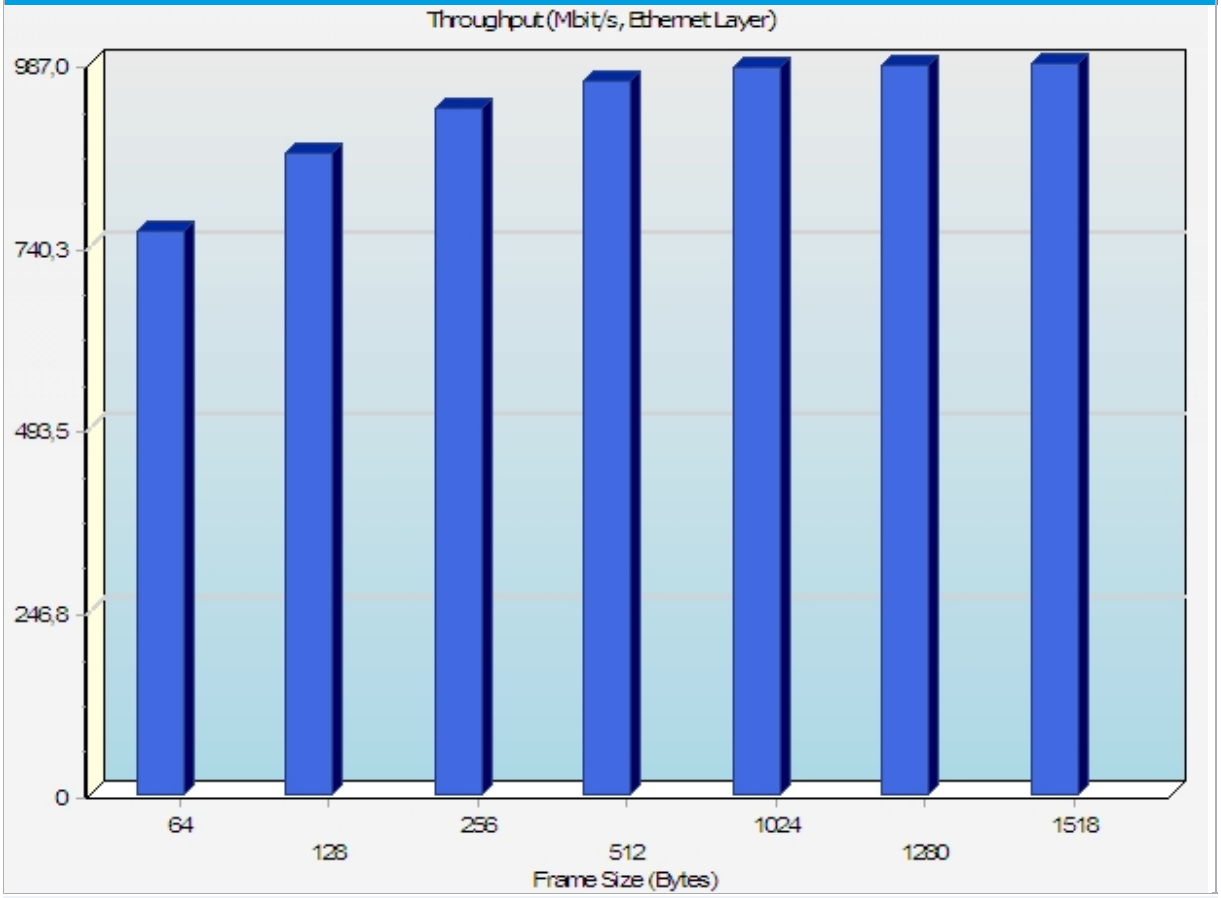
Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)				
		Layer		
		All	**Ethernet**	**IP**
64	**Min.**	1000,000	761,904	547,619
	Max.	1000,000	761,904	547,619
	Avg.	1000,000	761,904	547,619
128	**Min.**	1000,000	864,864	743,243
	Max.	1000,000	864,864	743,243
	Avg.	1000,000	864,864	743,243
256	**Min.**	1000,000	927,536	862,318
	Max.	1000,000	927,536	862,318
	Avg.	1000,000	927,536	862,318
512	**Min.**	1000,000	962,406	928,571
	Max.	1000,000	962,406	928,571
	Avg.	1000,000	962,406	928,571
1024	**Min.**	1000,000	980,842	963,601
	Max.	1000,000	980,842	963,601
	Avg.	1000,000	980,842	963,601
1280	**Min.**	1000,000	984,615	970,769
	Max.	1000,000	984,615	970,769
	Avg.	1000,000	984,615	970,769
1518	**Min.**	1000,000	986,996	975,292
	Max.	1000,000	986,996	975,292
	Avg.	1000,000	986,996	975,292

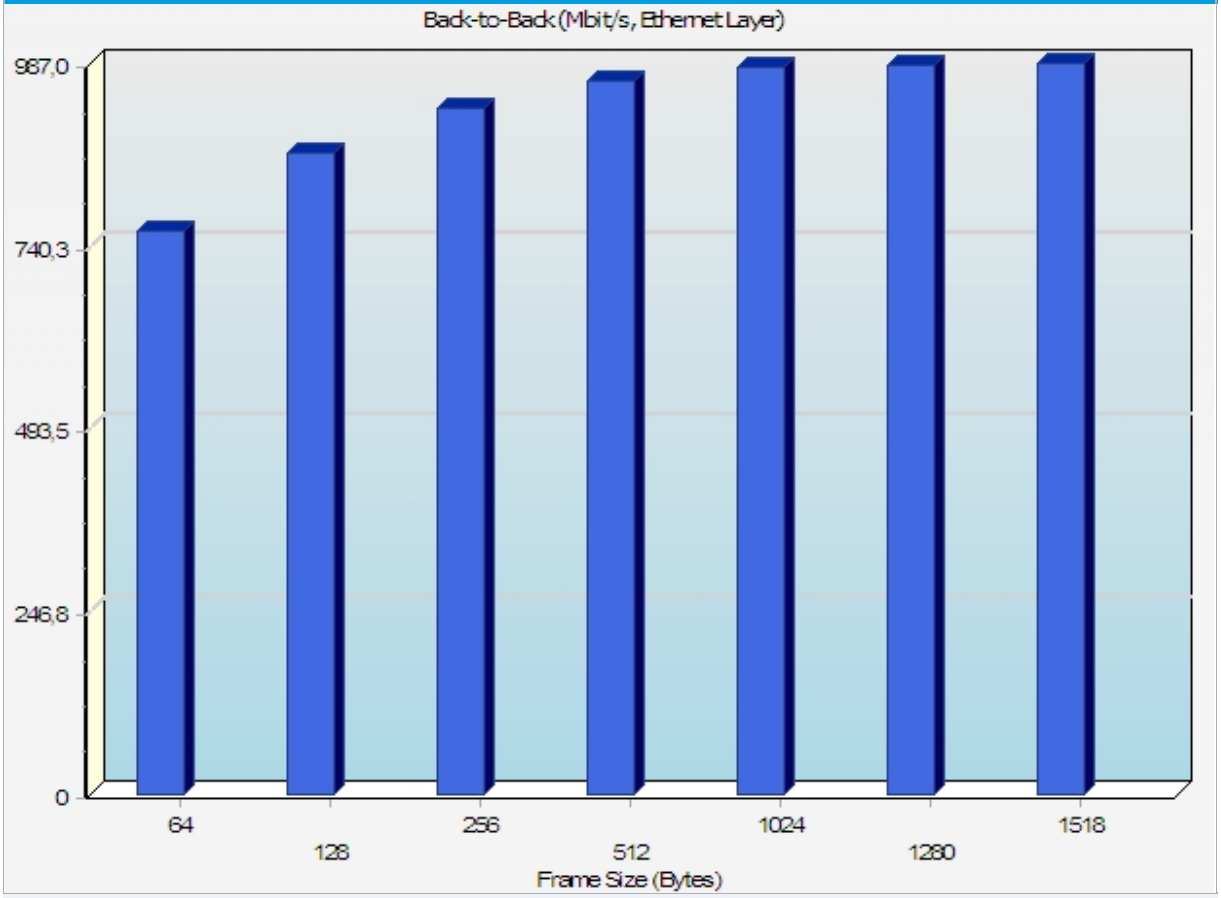
Back-to-Back (Mbit/s)				
		Layer		
		All	Ethernet	IP
64	Min.	1000,000	761,904	547,619
	Max.	1000,000	761,904	547,619
	Avg.	1000,000	761,904	547,619
128	Min.	1000,000	864,864	743,243
	Max.	1000,000	864,864	743,243
	Avg.	1000,000	864,864	743,243
256	Min.	1000,000	927,536	862,318
	Max.	1000,000	927,536	862,318
	Avg.	1000,000	927,536	862,318
512	Min.	1000,000	962,406	928,571
	Max.	1000,000	962,406	928,571
	Avg.	1000,000	962,406	928,571
1024	Min.	1000,000	980,842	963,601
	Max.	1000,000	980,842	963,601
	Avg.	1000,000	980,842	963,601
1280	Min.	1000,000	984,615	970,769
	Max.	1000,000	984,615	970,769
	Avg.	1000,000	984,615	970,769
1518	Min.	1000,000	986,996	975,292
	Max.	1000,000	986,996	975,292
	Avg.	1000,000	986,996	975,292

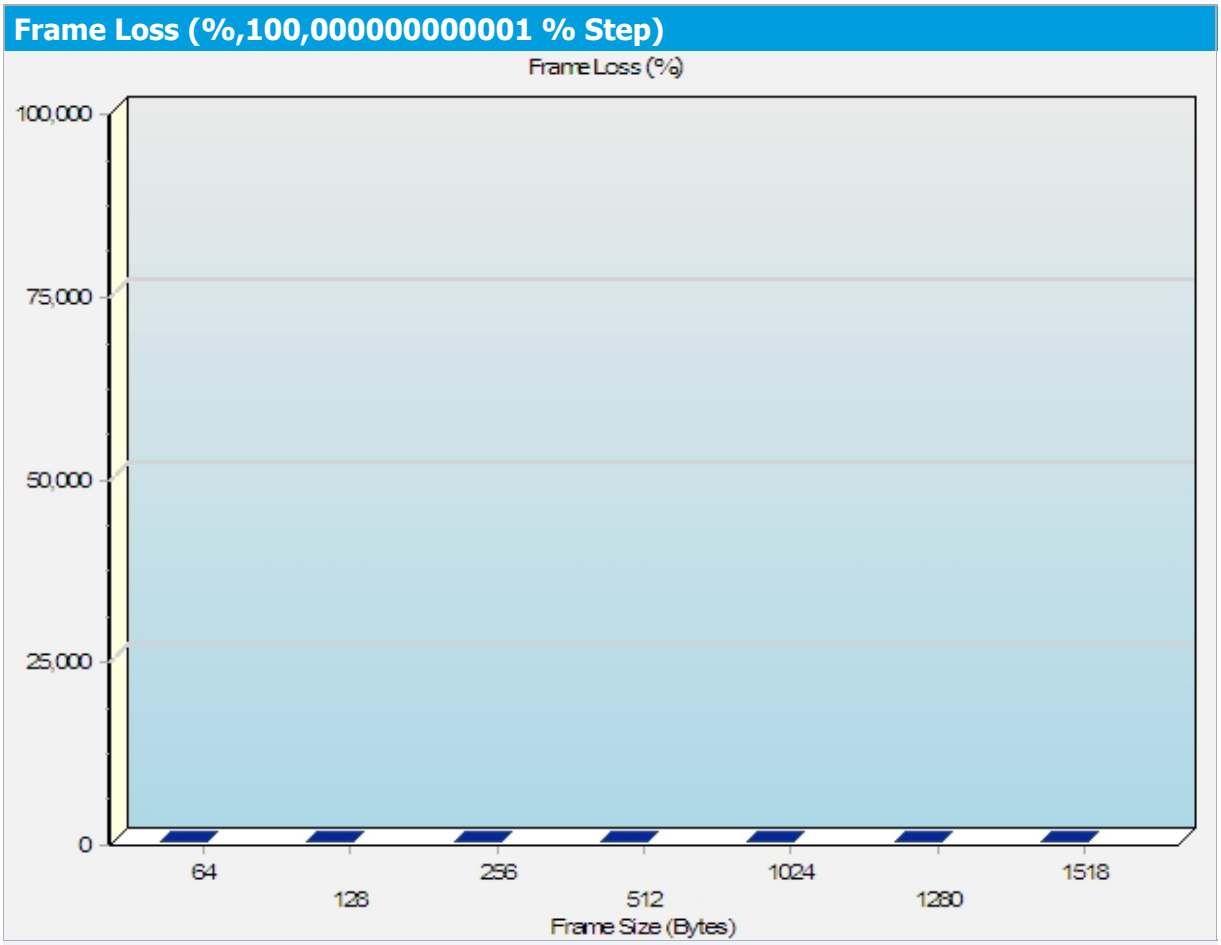
		Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	**Min.**	0,000	10,35
	Max.	0,000	10,50
	Avg.	0,000	10,46
128	**Min.**	0,000	10,91
	Max.	0,000	11,07
	Avg.	0,000	10,99
256	**Min.**	0,000	11,95
	Max.	0,000	12,10
	Avg.	0,000	12,02
512	**Min.**	0,000	14,02
	Max.	0,000	14,12
	Avg.	0,000	14,07
1024	**Min.**	0,000	18,15
	Max.	0,000	18,20
	Avg.	0,000	18,17
1280	**Min.**	0,000	20,11
	Max.	0,000	20,22
	Avg.	0,000	20,17
1518	**Min.**	0,000	22,07
	Max.	0,000	22,12
	Avg.	0,000	22,09

Throughput

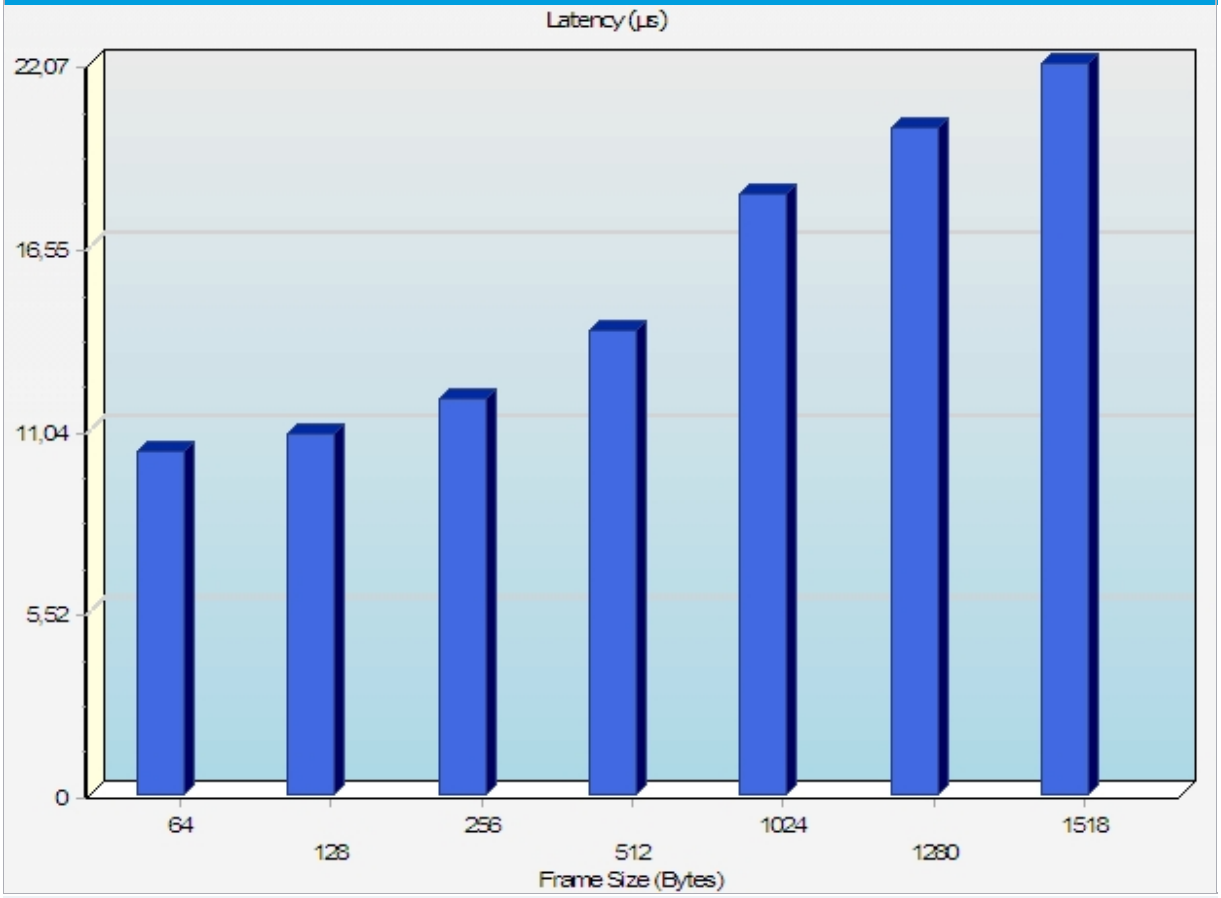

Throughput (Mbit/s, Ethernet Layer)
987,0
740,3
493,5
246,8
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

Back-to-Back


Back-to-Back (Mbit/s, Ethernet Layer)
987,0
740,3
493,5
246,8
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)


Frame Loss (%,100,000000000001 % Step)
Frame Loss (%)
100,000
75,000
50,000
25,000
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

Latency (µs, Store And Forward Mode)



TEST SETUP

Clock Synchronization	
Clock Mode	Recovered

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests						
Throughput	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Accuracy (%)	Acceptable Errors	Validations
	100,0000	00:01	10	1,00	0	1
Back-to-Back		Burst Time (MM:SS)	Trials	Accuracy (Frames)	Acceptable Errors	Bursts
		00:01	10	1	0	1
Frame Loss	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Granularity (%)		
	100,0000	00:01	10	10		
Latency	Max. Rate (%)	Trial Duration (MM:SS)	Trials			
	100,0000	00:01	10			

FUNCTIONS

Ping & Trace Route

Network	
Source MAC Address	00:03:01:FE:01:19
Source IP Address	10.10.1.25
Destination IP Address	0.0.0.0

Ping	
Timeout (ms)	4000
Delay (ms)	1000
Data Size (Bytes)	32
TTL	128
Type Of Service (TOS)	0x00

Ping Statistics	
Packets Transmitted	--
Packets Received	--
Percentage Lost (%)	--
Minimum Round Trip Time (ms)	--
Maximum Round Trip Time (ms)	--
Average Round Trip Time (ms)	--

Trace Route	
Timeout (ms)	4000
Max Hop Count	128

SYSTEM INFORMATION

Product Name	NetBlazer Series
Version	NetBlazer Series 2.8
Module ID	FTB-860
Slot ID	0
Software Product Version	2.8.0.28
	NetBlazer Series 2.8
Assembly Hardware Revision	D
Serial Number	707954
Calibration Date	28/04/2013 22:04:00

Software Options

100optical	Ethernet 100Base-FX (Optical) Interface	Disabled
GigE_Optical	Ethernet 1000Base-X (Optical) Interface	Enabled
GigE_Electrical	Ethernet 1000Base-T (Electrical) Interface	Enabled
FC-1X	Fibre Channel 1X	Disabled
FC-2X	Fibre Channel 2X	Disabled
TCP-THPUT	Tcp Throughput Test Application	Disabled
FC-4X	Fibre Channel 4X	Disabled
MPLS	MPLS Encapsulation (10GE and less)	Disabled
IPV6	Internet Protocol Version 6 (IPv6) (10GE and less)	Enabled
ETH-THRU	Through Mode Test Application	Enabled
Cable_Test	Cable Test	Disabled
TRAFFIC_GEN	Traffic Generation and Monitoring Test Application	Enabled
CPRI-1.2G	CPRI 1.2288 Gbit/s	Disabled
CPRI	CPRI 2.4576 Gbit/s and 3.072 Gbit/s	Disabled
CPRI-4.9G	CPRI 4.9152 Gbit/s	Disabled
CPRI-6.1G	CPRI 6.144 Gbit/s	Disabled
OBSAI	OBSAI 3.072 Gbit/s	Disabled
1588PTP	1588 Precision Time Protocol	Enabled
SyncE	Synchronous Ethernet	Disabled
ETH-OAM	Carrier Ethernet OAM Test Application	Disabled
ETH-CAPTURE	Ethernet Frame Capture for Advanced Troubleshooting	Disabled
ADV-FILTERS	Advanced Filtering	Disabled
RFC6349	RFC 6349 Test Application (10GE and less)	Disabled
iSAM	Intelligent Service Activation Methodology	Disabled
LINK-OAM	Link OAM	Disabled
TRAFFIC-SCAN	Traffic Scan	Disabled

LOGGER

ID	Time	Event	Duration	Details
1	11/11 09:56:43	Test Started		2016-11-11
2	11/11 10:57:42	Test Stopped		PASS

B.2. RFC 2544 test - 100Mbps

FTB-860 Report

| Managed Ethernet Switch IP - RFC 2544 |

PASS

Report Header:	System-on-Chip Engineering S.L.
Report Title:	Managed Ethernet Switch IP - RFC 2544
Report Date:	11/11/2016 9:54:20
Type:	RFC 2544

Job Information

Job ID:	
Circuit ID:	
Contractor Name:	
Customer Name:	
Operator Name:	Unknown Operator
Comment:	100Mbps Test

Table of contents

SUMMARY

Results Summary

Test Status	
Throughput	Completed 00d:00:17:44
Back-to-Back	Completed 00d:00:10:50
Frame Loss	Completed 00d:00:21:40
Latency	Completed 00d:00:10:49
Pass/Fail Verdict	PASS
Start Time	11/11/2016 08:35:34
Power Recovery	0

RFC 2544

Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)			
	Layer		
	All	**Ethernet**	**IP**
64	100,0000	76,1904	54,7619
128	100,0000	86,4864	74,3243
256	100,0000	92,7536	86,2318
512	100,0000	96,2406	92,8571
1024	100,0000	98,0842	96,3601
1280	100,0000	98,4615	97,0769
1518	100,0000	98,6996	97,5292

Back-to-Back (Mbit/s)			
	Layer		
	All	Ethernet	IP
64	100,0000	76,1904	54,7619
128	100,0000	86,4864	74,3243
256	100,0000	92,7536	86,2318
512	100,0000	96,2406	92,8571
1024	100,0000	98,0842	96,3601
1280	100,0000	98,4615	97,0769
1518	100,0000	98,6996	97,5292

	Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	0,000	79,73
128	0,000	84,89
256	0,000	95,24
512	0,000	115,75
1024	0,000	156,54
1280	0,000	176,96
1518	0,000	196,17

RX Frequency		
	Max Negative Offset	Max Positive Offset
Frequency Offset (ppm)	0,0	7,0

Alarms/Errors

Alarms/Errors List	
Active/Historical Alarms/Errors	None

Interface	
Alarms	**Seconds**
Frequency	0

Clock Sync	
Alarm	**Seconds**
LOC	0

Ethernet	
Alarms	**Seconds**
Link Down	0

Errors	Seconds	Count	Rate
Symbol	0	0	0,00E00
False Carrier	0	0	0,00E00
Idle	0	0	0,00E00
FCS	0	0	0,00E00
Jabber	0	0	0,00E00
Alignment	0	0	0,00E00
Oversize	Disabled		
Runt	0	0	0,00E00
Undersize	0	0	0,00E00
Total		0	

IP/UDP/TCP			
Errors	**Seconds**	**Count**	**Rate**
IP Checksum	0	0	0,00E00
UDP Checksum	0	0	0,00E00
Total		0	

Setup Summary

Application Type	RFC 2544

Interface	**Port 1**
Interface/Rate	10/100/1000M Electrical
Connector	RJ45

Auto-Negotiation	Enabled
Speed	100 Mbit/s
Duplex	Full
Flow Control	None

Framing: MAC/IP/UDP

IP Version	IPv4

	MAC	**IP Address**	**UDP Port**
Source	00:03:01:FE:01:19	10.10.1.25 / 255.255.0.0	49184
Destination	00:03:01:FE:01:1A	10.10.1.25	7

EtherType	0x0800
Automatic IP (DHCP)	Disabled
TTL	128
TOS	0x00
Default Gateway	Disabled

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests			
Throughput	Max. Rate (%)	Threshold (%)	Accuracy (%)
	100,0000	100,0	1,00
Back-to-Back	Burst Time (MM:SS)	Threshold (%)	Accuracy (Frames)
	00:01	100,0	1
Frame Loss	Max. Rate (%)	Threshold (%)	Granularity (%)
	100,0000	0,1	10
Latency	Max. Rate (%)		Threshold (ms)
	100,0000		125,0

TEST RESULTS

Ethernet Traffic

Frame Type	TX Count	RX Count
Multicast	0	0
Broadcast	420	0
Unicast	19574510	19574510
Non-Unicast	420	0
Total	19574930	19574510

Frame Size (Bytes)	RX Count	%
< 64	0	0
64	8779790	44
65 – 127	0	0
128 – 255	4983100	25
256 – 511	2672110	13
512 – 1023	1386280	7
1024 – 1518	1753230	8
> 1518	0	0
Total	19574510	

Flow Control Monitoring

Frame Count	
Pause Frames	0
Abort Frames	0
Frames RX	0
Frames TX	0

Pause Time (Quanta)	
Total Pause Time	0
Last Pause Time	0
Minimum Pause Time	0
Maximum Pause Time	0

RFC 2544

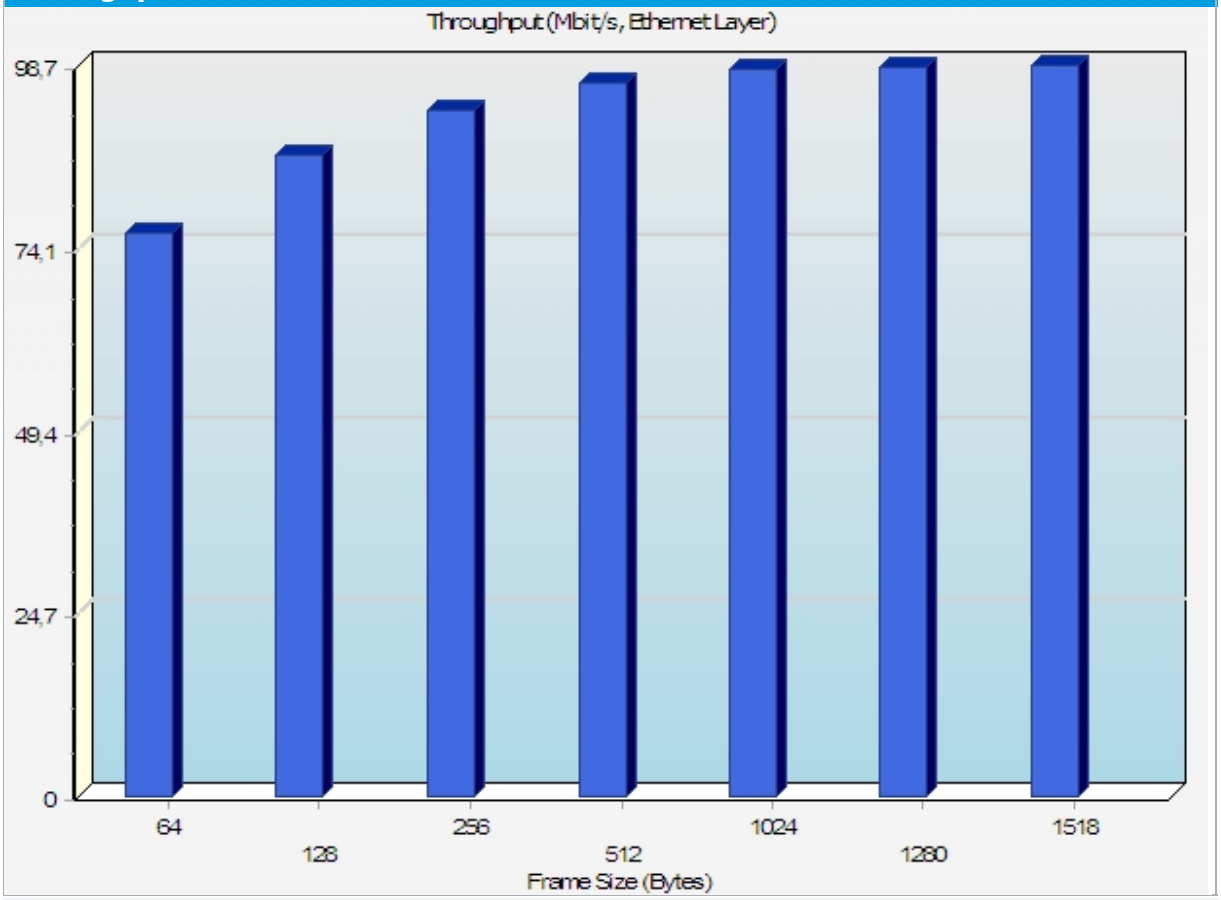
Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)				
		Layer		
		All	**Ethernet**	**IP**
64	**Min.**	100,0000	76,1904	54,7619
	Max.	100,0000	76,1904	54,7619
	Avg.	100,0000	76,1904	54,7619
128	**Min.**	100,0000	86,4864	74,3243
	Max.	100,0000	86,4864	74,3243
	Avg.	100,0000	86,4864	74,3243
256	**Min.**	100,0000	92,7536	86,2318
	Max.	100,0000	92,7536	86,2318
	Avg.	100,0000	92,7536	86,2318
512	**Min.**	100,0000	96,2406	92,8571
	Max.	100,0000	96,2406	92,8571
	Avg.	100,0000	96,2406	92,8571
1024	**Min.**	100,0000	98,0842	96,3601
	Max.	100,0000	98,0842	96,3601
	Avg.	100,0000	98,0842	96,3601
1280	**Min.**	100,0000	98,4615	97,0769
	Max.	100,0000	98,4615	97,0769
	Avg.	100,0000	98,4615	97,0769
1518	**Min.**	100,0000	98,6996	97,5292
	Max.	100,0000	98,6996	97,5292
	Avg.	100,0000	98,6996	97,5292

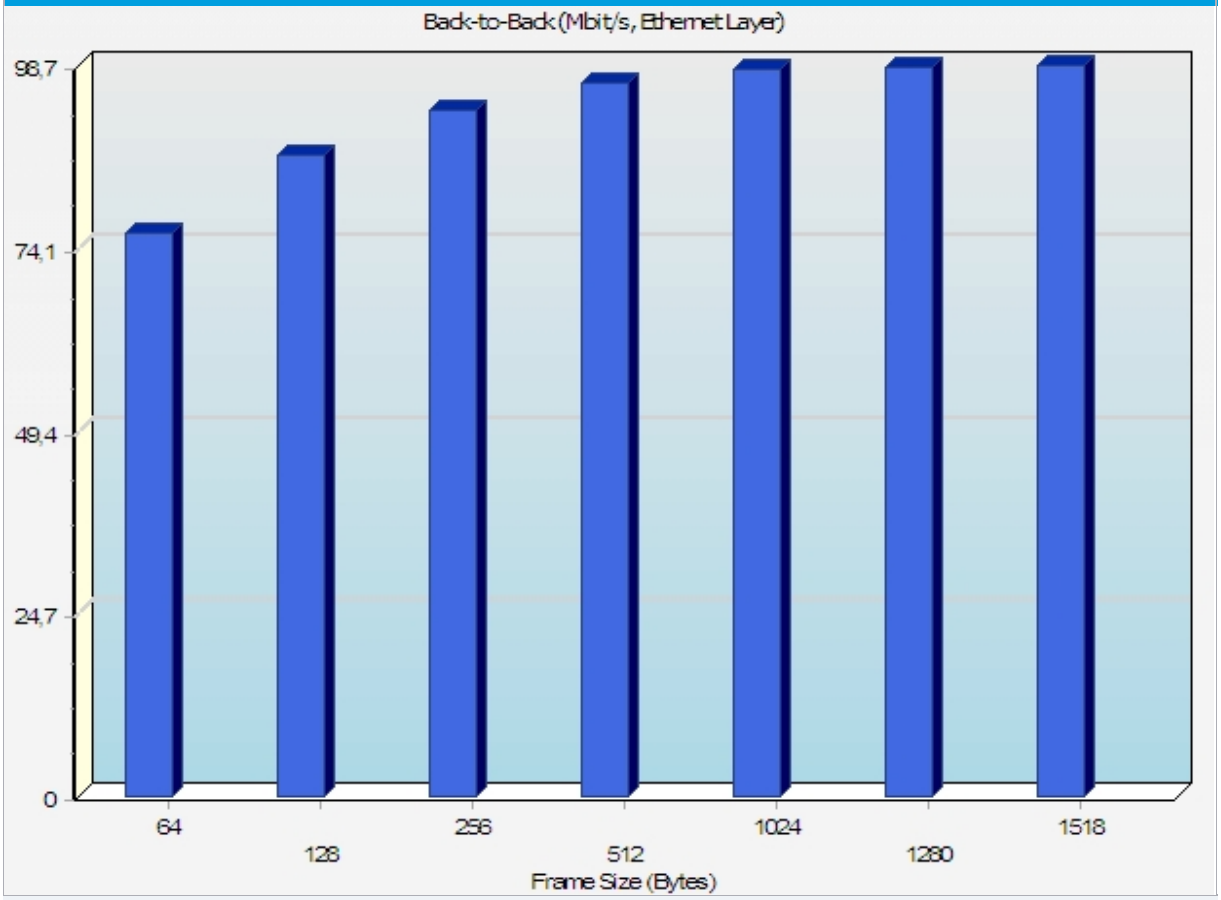
Back-to-Back (Mbit/s)				
		Layer		
		All	Ethernet	IP
64	Min.	100,0000	76,1904	54,7619
	Max.	100,0000	76,1904	54,7619
	Avg.	100,0000	76,1904	54,7619
128	Min.	100,0000	86,4864	74,3243
	Max.	100,0000	86,4864	74,3243
	Avg.	100,0000	86,4864	74,3243
256	Min.	100,0000	92,7536	86,2318
	Max.	100,0000	92,7536	86,2318
	Avg.	100,0000	92,7536	86,2318
512	Min.	100,0000	96,2406	92,8571
	Max.	100,0000	96,2406	92,8571
	Avg.	100,0000	96,2406	92,8571
1024	Min.	100,0000	98,0842	96,3601
	Max.	100,0000	98,0842	96,3601
	Avg.	100,0000	98,0842	96,3601
1280	Min.	100,0000	98,4615	97,0769
	Max.	100,0000	98,4615	97,0769
	Avg.	100,0000	98,4615	97,0769
1518	Min.	100,0000	98,6996	97,5292
	Max.	100,0000	98,6996	97,5292
	Avg.	100,0000	98,6996	97,5292

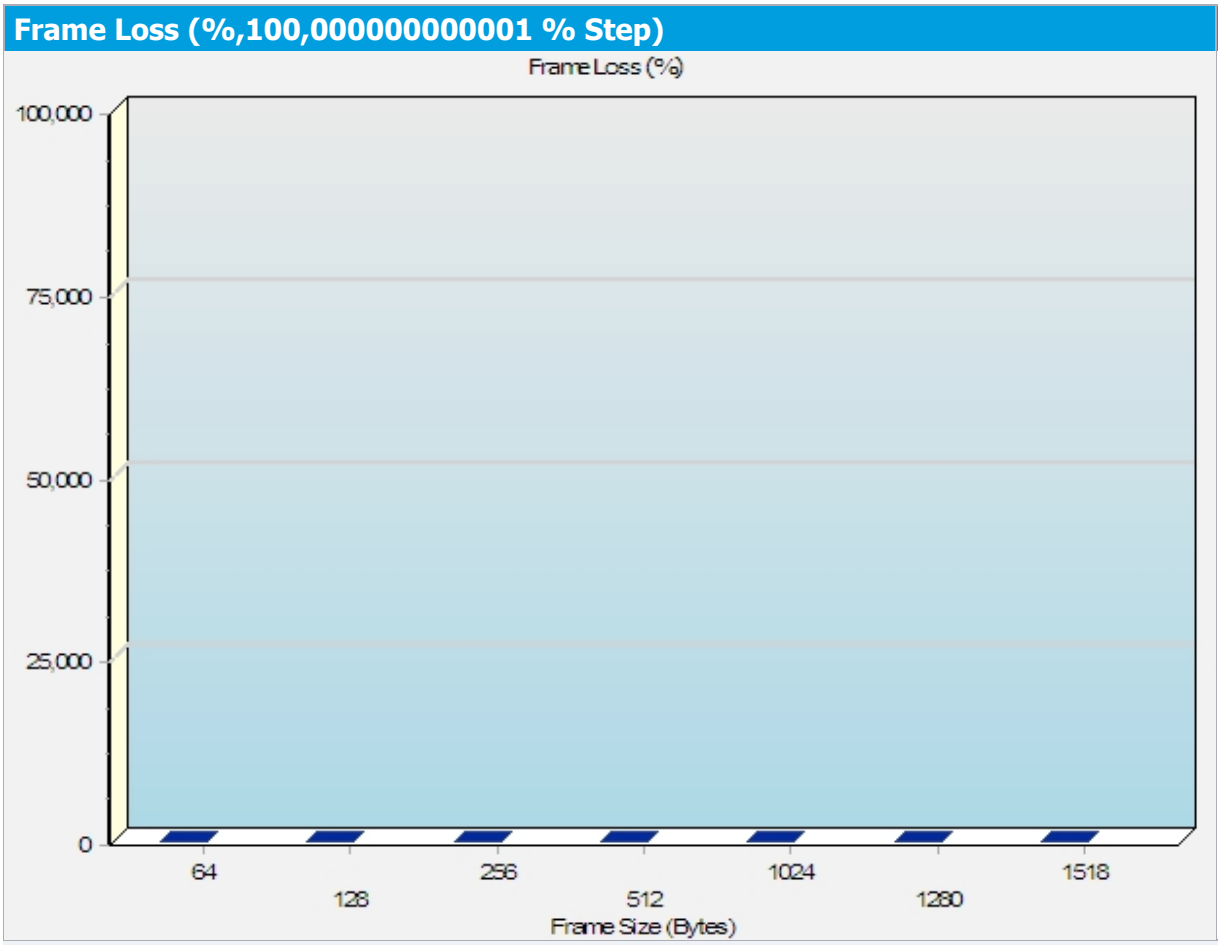
		Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	**Min.**	0,000	79,45
	Max.	0,000	79,86
	Avg.	0,000	79,73
128	**Min.**	0,000	84,47
	Max.	0,000	85,34
	Avg.	0,000	84,89
256	**Min.**	0,000	94,85
	Max.	0,000	95,37
	Avg.	0,000	95,24
512	**Min.**	0,000	115,27
	Max.	0,000	116,14
	Avg.	0,000	115,75
1024	**Min.**	0,000	156,20
	Max.	0,000	156,87
	Avg.	0,000	156,54
1280	**Min.**	0,000	176,67
	Max.	0,000	177,23
	Avg.	0,000	176,96
1518	**Min.**	0,000	195,90
	Max.	0,000	196,57
	Avg.	0,000	196,17

Throughput

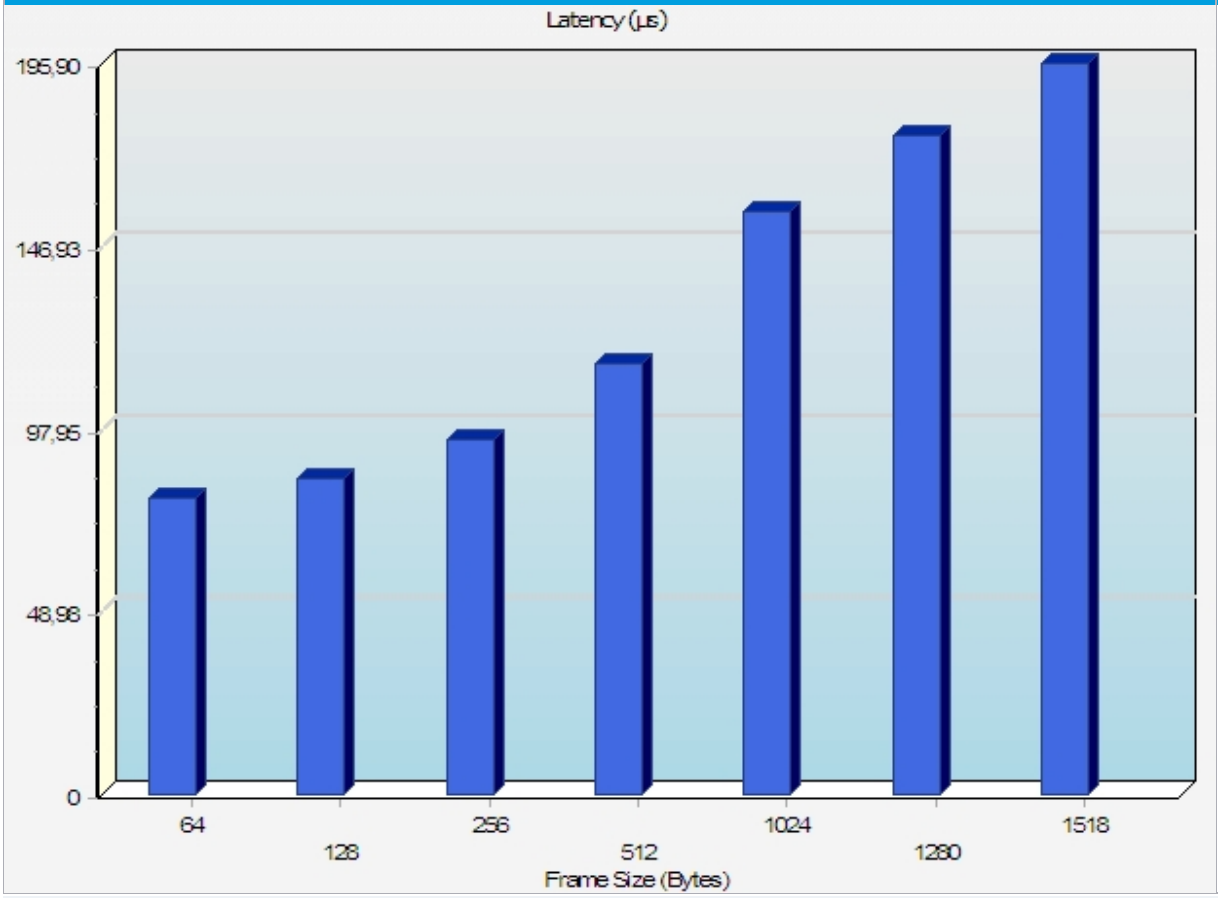

Throughput (Mbit/s, Ethernet Layer)
98,7
74,1
49,4
24,7
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

Back-to-Back


Back-to-Back (Mbit/s, Ethernet Layer)
98,7
74,1
49,4
24,7
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)


Frame Loss (%,100,000000000001 % Step)
Frame Loss (%)
100,000
75,000
50,000
25,000
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

Latency (µs, Store And Forward Mode)



TEST SETUP

Clock Synchronization	
Clock Mode	Internal

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests						
Throughput	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Accuracy (%)	Acceptable Errors	Validations
	100,0000	00:01	10	1,00	0	1
Back-to-Back		Burst Time (MM:SS)	Trials	Accuracy (Frames)	Acceptable Errors	Bursts
		00:01	10	1	0	1
Frame Loss	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Granularity (%)		
	100,0000	00:01	10	10		
Latency	Max. Rate (%)	Trial Duration (MM:SS)	Trials			
	100,0000	00:01	10			

FUNCTIONS

Ping & Trace Route

Network	
Source MAC Address	00:03:01:FE:01:19
Source IP Address	10.10.1.25
Destination IP Address	0.0.0.0

Ping	
Timeout (ms)	4000
Delay (ms)	1000
Data Size (Bytes)	32
TTL	128
Type Of Service (TOS)	0x00

Ping Statistics	
Packets Transmitted	--
Packets Received	--
Percentage Lost (%)	--
Minimum Round Trip Time (ms)	--
Maximum Round Trip Time (ms)	--
Average Round Trip Time (ms)	--

Trace Route	
Timeout (ms)	4000
Max Hop Count	128

SYSTEM INFORMATION

Product Name	NetBlazer Series
Version	NetBlazer Series 2.8
Module ID	FTB-860
Slot ID	0
Software Product Version	2.8.0.28
	NetBlazer Series 2.8
Assembly Hardware Revision	D
Serial Number	707954
Calibration Date	28/04/2013 22:04:00

Software Options

100optical	Ethernet 100Base-FX (Optical) Interface	Disabled
GigE_Optical	Ethernet 1000Base-X (Optical) Interface	Enabled
GigE_Electrical	Ethernet 1000Base-T (Electrical) Interface	Enabled
FC-1X	Fibre Channel 1X	Disabled
FC-2X	Fibre Channel 2X	Disabled
TCP-THPUT	Tcp Throughput Test Application	Disabled
FC-4X	Fibre Channel 4X	Disabled
MPLS	MPLS Encapsulation (10GE and less)	Disabled
IPV6	Internet Protocol Version 6 (IPv6) (10GE and less)	Enabled
ETH-THRU	Through Mode Test Application	Enabled
Cable_Test	Cable Test	Disabled
TRAFFIC_GEN	Traffic Generation and Monitoring Test Application	Enabled
CPRI-1.2G	CPRI 1.2288 Gbit/s	Disabled
CPRI	CPRI 2.4576 Gbit/s and 3.072 Gbit/s	Disabled
CPRI-4.9G	CPRI 4.9152 Gbit/s	Disabled
CPRI-6.1G	CPRI 6.144 Gbit/s	Disabled
OBSAI	OBSAI 3.072 Gbit/s	Disabled
1588PTP	1588 Precision Time Protocol	Enabled
SyncE	Synchronous Ethernet	Disabled
ETH-OAM	Carrier Ethernet OAM Test Application	Disabled
ETH-CAPTURE	Ethernet Frame Capture for Advanced Troubleshooting	Disabled
ADV-FILTERS	Advanced Filtering	Disabled
RFC6349	RFC 6349 Test Application (10GE and less)	Disabled
iSAM	Intelligent Service Activation Methodology	Disabled
LINK-OAM	Link OAM	Disabled
TRAFFIC-SCAN	Traffic Scan	Disabled

LOGGER

ID	Time	Event	Duration	Details
1	11/11 08:35:34	Test Started		2016-11-11
2	11/11 09:36:38	Test Stopped		PASS

B.3. RFC 2544 test - 10Mbps

FTB-860 Report

| Managed Ethernet Switch IP - RFC 2544 |

PASS

Report Header:	System-on-Chip Engineering S.L.
Report Title:	Managed Ethernet Switch IP - RFC 2544
Report Date:	11/11/2016 12:36:16
Type:	RFC 2544

Job Information

Job ID:	
Circuit ID:	
Contractor Name:	
Customer Name:	
Operator Name:	Unknown Operator
Comment:	10Mbps Test

Table of contents

SUMMARY

Results Summary

Test Status	
Throughput	Completed 00d:00:17:40
Back-to-Back	Completed 00d:00:10:47
Frame Loss	Completed 00d:00:21:39
Latency	Completed 00d:00:10:48
Pass/Fail Verdict	PASS
Start Time	11/11/2016 11:21:23
Power Recovery	0

RFC 2544

Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)			
	Layer		
	All	**Ethernet**	**IP**
64	10,00000	7,61904	5,47619
128	10,00000	8,64864	7,43243
256	10,00000	9,27536	8,62318
512	10,00000	9,62406	9,28571
1024	10,00000	9,80842	9,63601
1280	10,00000	9,84615	9,70769
1518	10,00000	9,86996	9,75292

Back-to-Back (Mbit/s)			
	Layer		
	All	Ethernet	IP
64	10,00000	7,61904	5,47619
128	10,00000	8,64864	7,43243
256	10,00000	9,27536	8,62318
512	10,00000	9,62406	9,28571
1024	10,00000	9,80842	9,63601
1280	10,00000	9,84615	9,70769
1518	10,00000	9,86996	9,75292

	Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	0,000	785,47
128	0,000	836,22
256	0,000	938,90
512	0,000	1143,74
1024	0,000	1554,22
1280	0,000	1759,00
1518	0,000	1948,27

RX Frequency		
	Max Negative Offset	Max Positive Offset
Frequency Offset (ppm)		

Alarms/Errors

Alarms/Errors List	
Active/Historical Alarms/Errors	None

Interface	
Alarms	**Seconds**

Clock Sync	
Alarm	**Seconds**
LOC	0

Ethernet	
Alarms	**Seconds**
Link Down	0

Errors	Seconds	Count	Rate
FCS	0	0	0,00E00
Jabber	0	0	0,00E00
Alignment	0	0	0,00E00
Oversize	Disabled		
Runt	0	0	0,00E00
Undersize	0	0	0,00E00
Total		0	

IP/UDP/TCP			
Errors	**Seconds**	**Count**	**Rate**
IP Checksum	0	0	0,00E00
UDP Checksum	0	0	0,00E00
Total		0	

Setup Summary

Application Type	RFC 2544

Interface	Port 1
Interface/Rate	10/100/1000M Electrical
Connector	RJ45

Auto-Negotiation	Enabled
Speed	10 Mbit/s
Duplex	Full
Flow Control	None

Framing: MAC/IP/UDP

IP Version	IPv4

	MAC	IP Address	UDP Port
Source	00:03:01:FE:01:19	10.10.1.25 / 255.255.0.0	49184
Destination	00:03:01:FE:01:1A	10.10.1.25	7

EtherType	0x0800
Automatic IP (DHCP)	Disabled
TTL	128
TOS	0x00
Default Gateway	Disabled

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests			
Throughput	Max. Rate (%)	Threshold (%)	Accuracy (%)
	100,0000	100,0	1,00
Back-to-Back	Burst Time (MM:SS)	Threshold (%)	Accuracy (Frames)
	00:01	100,0	1
Frame Loss	Max. Rate (%)	Threshold (%)	Granularity (%)
	100,0000	0,1	10
Latency	Max. Rate (%)		Threshold (ms)
	100,0000		125,0

TEST RESULTS

Ethernet Traffic

Frame Type	TX Count	RX Count
Multicast	0	0
Broadcast	420	0
Unicast	1957500	1957500
Non-Unicast	420	0
Total	1957920	1957500

Frame Size (Bytes)	RX Count	%
< 64	0	0
64	877980	44
65 – 127	0	0
128 – 255	498310	25
256 – 511	267210	13
512 – 1023	138650	7
1024 – 1518	175350	8
> 1518	0	0
Total	1957500	

Flow Control Monitoring

Frame Count	
Pause Frames	0
Abort Frames	0
Frames RX	0
Frames TX	0

Pause Time (Quanta)	
Total Pause Time	0
Last Pause Time	0
Minimum Pause Time	0
Maximum Pause Time	0

RFC 2544

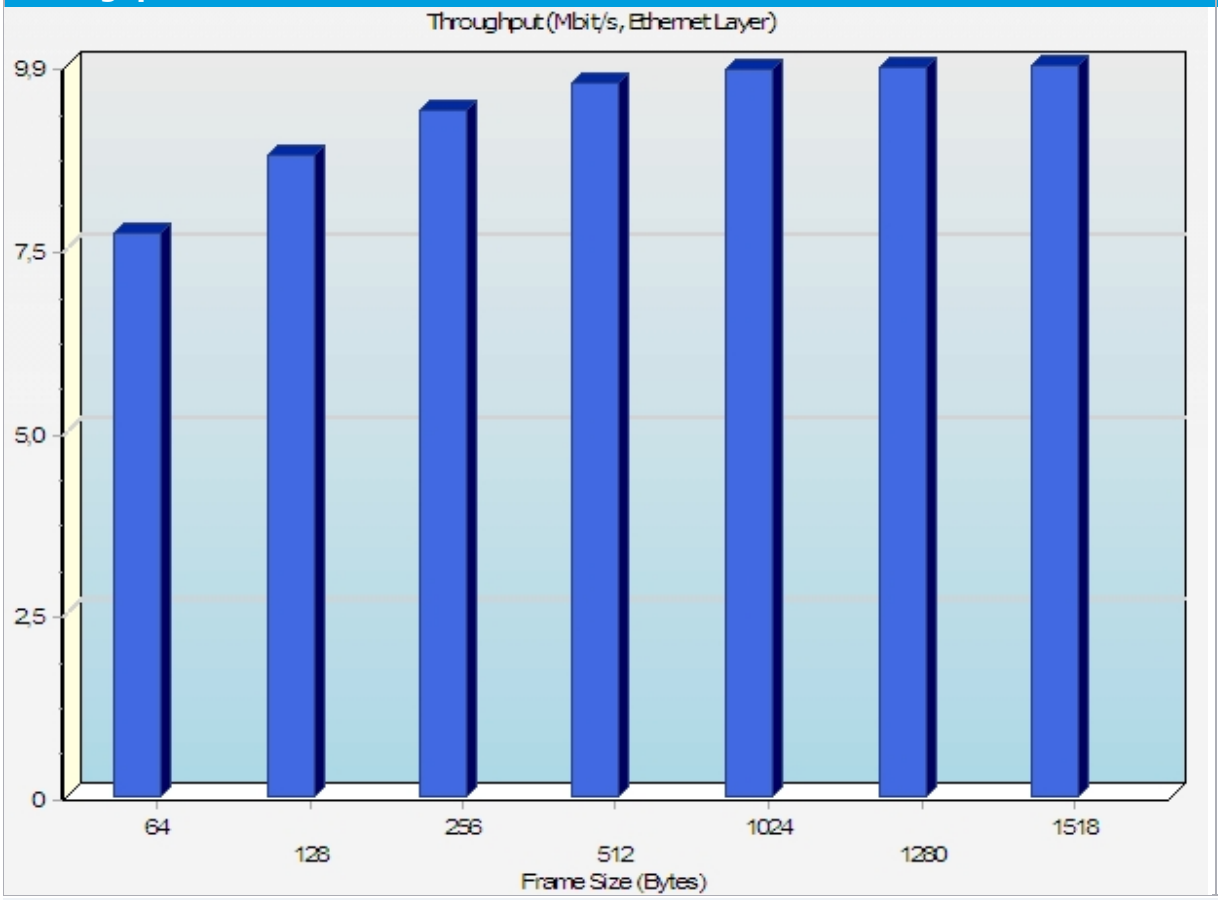
Pass/Fail Verdict	
Throughput	PASS
Back-to-Back	PASS
Frame Loss	PASS
Latency	PASS

Throughput (Mbit/s)				
		Layer		
		All	**Ethernet**	**IP**
64	**Min.**	10,00000	7,61904	5,47619
	Max.	10,00000	7,61904	5,47619
	Avg.	10,00000	7,61904	5,47619
128	**Min.**	10,00000	8,64864	7,43243
	Max.	10,00000	8,64864	7,43243
	Avg.	10,00000	8,64864	7,43243
256	**Min.**	10,00000	9,27536	8,62318
	Max.	10,00000	9,27536	8,62318
	Avg.	10,00000	9,27536	8,62318
512	**Min.**	10,00000	9,62406	9,28571
	Max.	10,00000	9,62406	9,28571
	Avg.	10,00000	9,62406	9,28571
1024	**Min.**	10,00000	9,80842	9,63601
	Max.	10,00000	9,80842	9,63601
	Avg.	10,00000	9,80842	9,63601
1280	**Min.**	10,00000	9,84615	9,70769
	Max.	10,00000	9,84615	9,70769
	Avg.	10,00000	9,84615	9,70769
1518	**Min.**	10,00000	9,86996	9,75292
	Max.	10,00000	9,86996	9,75292
	Avg.	10,00000	9,86996	9,75292

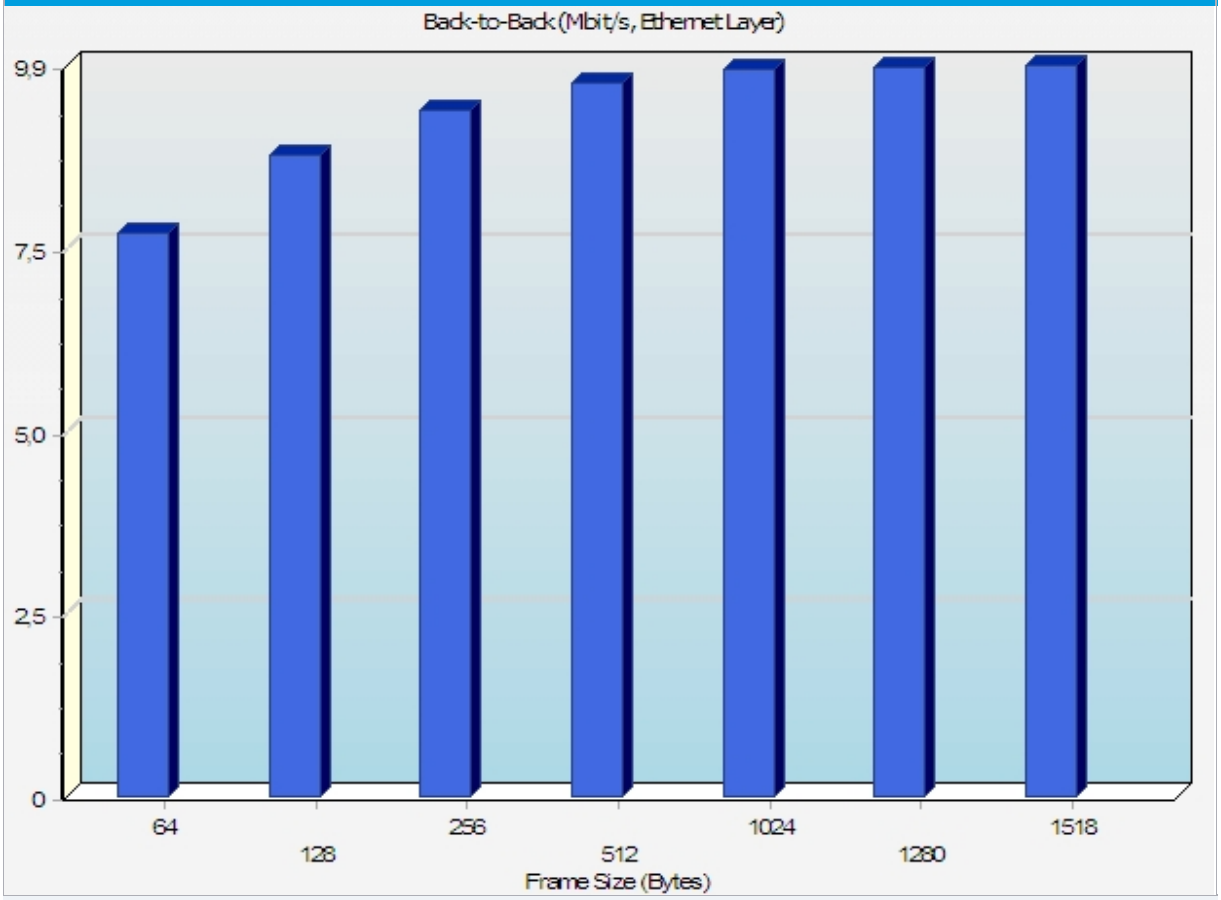
Back-to-Back (Mbit/s)				
		Layer		
		All	Ethernet	IP
64	Min.	10,00000	7,61904	5,47619
	Max.	10,00000	7,61904	5,47619
	Avg.	10,00000	7,61904	5,47619
128	Min.	10,00000	8,64864	7,43243
	Max.	10,00000	8,64864	7,43243
	Avg.	10,00000	8,64864	7,43243
256	Min.	10,00000	9,27536	8,62318
	Max.	10,00000	9,27536	8,62318
	Avg.	10,00000	9,27536	8,62318
512	Min.	10,00000	9,62406	9,28571
	Max.	10,00000	9,62406	9,28571
	Avg.	10,00000	9,62406	9,28571
1024	Min.	10,00000	9,80842	9,63601
	Max.	10,00000	9,80842	9,63601
	Avg.	10,00000	9,80842	9,63601
1280	Min.	10,00000	9,84615	9,70769
	Max.	10,00000	9,84615	9,70769
	Avg.	10,00000	9,84615	9,70769
1518	Min.	10,00000	9,86996	9,75292
	Max.	10,00000	9,86996	9,75292
	Avg.	10,00000	9,86996	9,75292

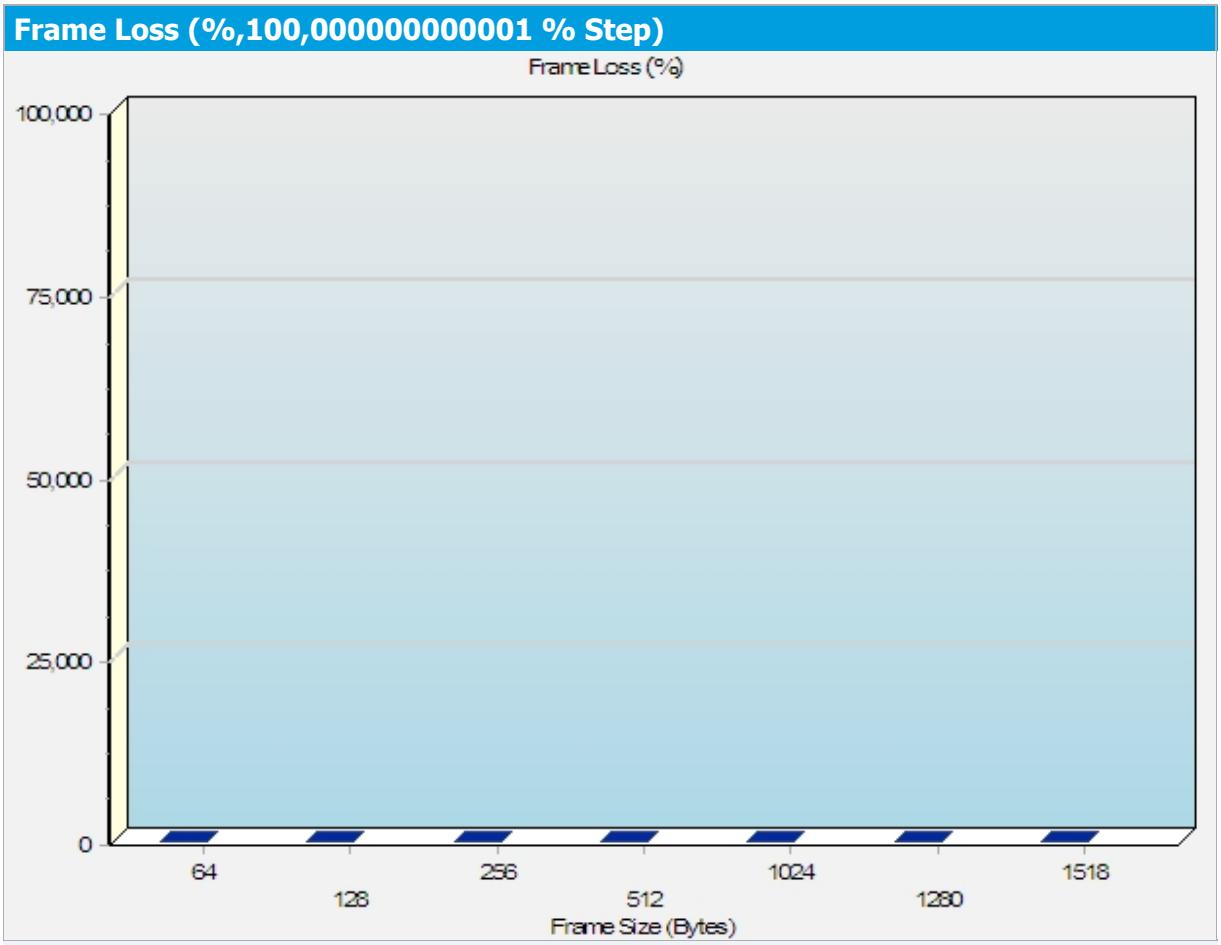
		Frame Loss (%,100,0000 % Step)	Latency (µs, S. & F.) Mode
64	**Min.**	0,000	780,24
	Max.	0,000	789,55
	Avg.	0,000	785,47
128	**Min.**	0,000	832,07
	Max.	0,000	841,02
	Avg.	0,000	836,22
256	**Min.**	0,000	934,04
	Max.	0,000	943,20
	Avg.	0,000	938,90
512	**Min.**	0,000	1139,22
	Max.	0,000	1148,38
	Avg.	0,000	1143,74
1024	**Min.**	0,000	1548,71
	Max.	0,000	1557,81
	Avg.	0,000	1554,22
1280	**Min.**	0,000	1754,61
	Max.	0,000	1762,68
	Avg.	0,000	1759,00
1518	**Min.**	0,000	1945,28
	Max.	0,000	1951,09
	Avg.	0,000	1948,27

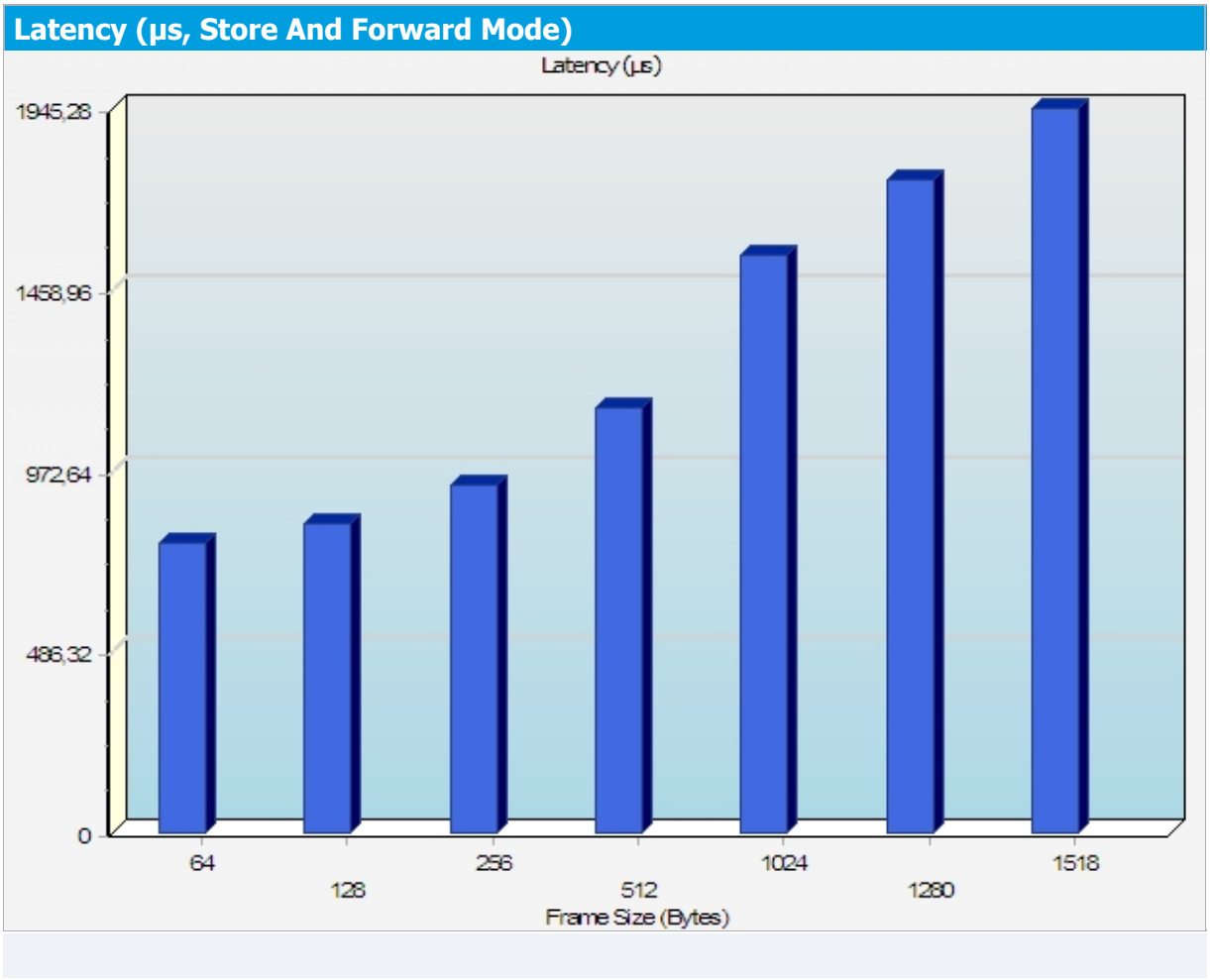
Throughput


Throughput (Mbit/s, Ethernet Layer)
9,9
7,5
5,0
2,5
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

Back-to-Back


Back-to-Back (Mbit/s, Ethernet Layer)
9,9
7,5
5,0
2,5
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)


Frame Loss (%,100,000000000001 % Step)
Frame Loss (%)
100,000
75,000
50,000
25,000
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)


Latency (µs, Store And Forward Mode)
Latency (µs)
1945,28
1458,96
972,64
486,32
0
64
128
256
512
1024
1280
1518
Frame Size (Bytes)

TEST SETUP

Clock Synchronization	
Clock Mode	Internal

RFC 2544

Frame Size Distribution (Bytes)						
64	128	256	512	1024	1280	1518

Enabled Subtests						
Throughput	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Accuracy (%)	Acceptable Errors	Validations
	100,0000	00:01	10	1,00	0	1
Back-to-Back		Burst Time (MM:SS)	Trials	Accuracy (Frames)	Acceptable Errors	Bursts
		00:01	10	1	0	1
Frame Loss	Max. Rate (%)	Trial Duration (MM:SS)	Trials	Granularity (%)		
	100,0000	00:01	10	10		
Latency	Max. Rate (%)	Trial Duration (MM:SS)	Trials			
	100,0000	00:01	10			

FUNCTIONS

Ping & Trace Route

Network	
Source MAC Address	00:03:01:FE:01:19
Source IP Address	10.10.1.25
Destination IP Address	0.0.0.0

Ping	
Timeout (ms)	4000
Delay (ms)	1000
Data Size (Bytes)	32
TTL	128
Type Of Service (TOS)	0x00

Ping Statistics	
Packets Transmitted	--
Packets Received	--
Percentage Lost (%)	--
Minimum Round Trip Time (ms)	--
Maximum Round Trip Time (ms)	--
Average Round Trip Time (ms)	--

Trace Route	
Timeout (ms)	4000
Max Hop Count	128

SYSTEM INFORMATION

Product Name	NetBlazer Series
Version	NetBlazer Series 2.8
Module ID	FTB-860
Slot ID	0
Software Product Version	2.8.0.28
	NetBlazer Series 2.8
Assembly Hardware Revision	D
Serial Number	707954
Calibration Date	28/04/2013 22:04:00

Software Options

100optical	Ethernet 100Base-FX (Optical) Interface	Disabled
GigE_Optical	Ethernet 1000Base-X (Optical) Interface	Enabled
GigE_Electrical	Ethernet 1000Base-T (Electrical) Interface	Enabled
FC-1X	Fibre Channel 1X	Disabled
FC-2X	Fibre Channel 2X	Disabled
TCP-THPUT	Tcp Throughput Test Application	Disabled
FC-4X	Fibre Channel 4X	Disabled
MPLS	MPLS Encapsulation (10GE and less)	Disabled
IPV6	Internet Protocol Version 6 (IPv6) (10GE and less)	Enabled
ETH-THRU	Through Mode Test Application	Enabled
Cable_Test	Cable Test	Disabled
TRAFFIC_GEN	Traffic Generation and Monitoring Test Application	Enabled
CPRI-1.2G	CPRI 1.2288 Gbit/s	Disabled
CPRI	CPRI 2.4576 Gbit/s and 3.072 Gbit/s	Disabled
CPRI-4.9G	CPRI 4.9152 Gbit/s	Disabled
CPRI-6.1G	CPRI 6.144 Gbit/s	Disabled
OBSAI	OBSAI 3.072 Gbit/s	Disabled
1588PTP	1588 Precision Time Protocol	Enabled
SyncE	Synchronous Ethernet	Disabled
ETH-OAM	Carrier Ethernet OAM Test Application	Disabled
ETH-CAPTURE	Ethernet Frame Capture for Advanced Troubleshooting	Disabled
ADV-FILTERS	Advanced Filtering	Disabled
RFC6349	RFC 6349 Test Application (10GE and less)	Disabled
iSAM	Intelligent Service Activation Methodology	Disabled
LINK-OAM	Link OAM	Disabled
TRAFFIC-SCAN	Traffic Scan	Disabled

LOGGER

ID	Time	Event	Duration	Details
1	11/11 11:21:23	Test Started		2016-11-11
2	11/11 12:22:20	Test Stopped		PASS

C. Appendix C

This appendix contains the complete report of the *BER* test created by *EXFO FTB-1* tester.

C.1. BER test

FTB-860 Report

| Managed Ethernet Switch IP - BER Test |

PASS

Report Header:	System-on-Chip Engineering S.L.
Report Title:	Managed Ethernet Switch IP - BER Test
Report Date:	11/11/2016 14:52:08
Type:	EtherBERT - Framed

Job Information

Job ID:	
Circuit ID:	
Contractor Name:	
Customer Name:	
Operator Name:	Unknown Operator
Comment:	1000 Mpbs test

Table of contents

SUMMARY

Results Summary

Test Status	
Test Status	Completed
Pass/Fail Verdict	PASS
Start Time	11/11/2016 12:56:12
Duration	00d:01:00:03
Power Recovery	0

BER Pass/Fail Verdict	
Pass/Fail Verdict	PASS

BER Alarms	Seconds
No Traffic	0
Pattern Loss	0

BER Errors	Seconds	Count	Rate
Bit Error	0	0	0,00E00
Mismatch '0'	0	0	0,00E00
Mismatch '1'	0	0	0,00E00

Traffic	Frame Count
Total TX	441070751
Total RX	441070751

Service Disruption						
Longest (ms)	**Shortest (ms)**	**Last (ms)**	**Average (ms)**	**Total duration (ms)**	**Count**	**Verdict**
0,000	0,000	0,000	0,000	0,000	0	Disabled

RX Frequency		
	Max Negative Offset	**Max Positive Offset**
Frequency Offset (ppm)	0,0	5,1

Alarms/Errors

Alarms/Errors List	
Active/Historical Alarms/Errors	None

Interface	
Alarms	**Seconds**
Frequency	0

Clock Sync	
Alarm	**Seconds**
LOC	0

Ethernet	
Alarms	**Seconds**
Link Down	0

Errors	Seconds	Count	Rate
Symbol	0	0	0,00E00
False Carrier	0	0	0,00E00
Idle	0	0	0,00E00
FCS	0	0	0,00E00
Jabber	0	0	0,00E00
Oversize	0	0	0,00E00
Runt	0	0	0,00E00
Undersize	0	0	0,00E00
Total		0	

BER Alarms	Seconds
No Traffic	0
Pattern Loss	0

BER Errors	Seconds	Count	Rate
Bit Error	0	0	0,00E00
Mismatch '0'	0	0	0,00E00
Mismatch '1'	0	0	0,00E00

IP/UDP/TCP			
Errors	**Seconds**	**Count**	**Rate**
IP Checksum	0	0	0,00E00
UDP Checksum	0	0	0,00E00
TCP Checksum	0	0	0,00E00
Total		0	

Setup Summary

Application Type	EtherBERT - Framed

Interface	Port 1
Interface/Rate	10/100/1000M Electrical
Connector	RJ45

Auto-Negotiation	Enabled
Speed	1000 Mbit/s
Duplex	Full
Flow Control	None
Local Clock	Slave

Framing: MAC/IP/UDP

IP Version	IPv4

	MAC	IP Address	UDP Port
Source	00:03:01:FE:01:19	10.10.1.25 / 255.255.0.0	49184
Destination	00:03:01:FE:01:1A	10.10.1.25	7

EtherType	0x0800
Automatic IP (DHCP)	Disabled
TTL	128
TOS	0x00
Default Gateway	Disabled

BER

	TX	RX
Pattern	PRBS31 Non-Inverted	PRBS31 Non-Inverted

Bit Error	
Pass/Fail Verdict	Bit Error Rate
BER Threshold	1,0E-12

Service Disruption	
No Traffic Time (ms)	50,0
Pass/Fail Verdict	Disabled

Frame Size (Bytes)	1000 Fixed
TX Rate (%)	100,0000

TEST RESULTS

Ethernet Traffic

Frame Type	TX Count	RX Count
Multicast	0	0
Broadcast	0	0
Unicast	441070751	441070751
Non-Unicast	0	0
Total	441070751	441070751

Frame Size (Bytes)	RX Count	%
< 64	0	0
64	0	0
65 – 127	0	0
128 – 255	0	0
256 – 511	0	0
512 – 1023	441070751	100
1024 – 1518	0	0
> 1518	0	0
Total	441070751	

TEST SETUP

Clock Synchronization	
Clock Mode	Recovered

FUNCTIONS

Ping & Trace Route

Network	
Source MAC Address	00:03:01:FE:01:19
Source IP Address	10.10.1.25
Destination IP Address	0.0.0.0

Ping	
Timeout (ms)	4000
Delay (ms)	1000
Data Size (Bytes)	32
TTL	128
Type Of Service (TOS)	0x00

Ping Statistics	
Packets Transmitted	--
Packets Received	--
Percentage Lost (%)	--
Minimum Round Trip Time (ms)	--
Maximum Round Trip Time (ms)	--
Average Round Trip Time (ms)	--

Trace Route	
Timeout (ms)	4000
Max Hop Count	128

SYSTEM INFORMATION

Product Name	NetBlazer Series
Version	NetBlazer Series 2.8
Module ID	FTB-860
Slot ID	0
Software Product Version	2.8.0.28
	NetBlazer Series 2.8
Assembly Hardware Revision	D
Serial Number	707954
Calibration Date	28/04/2013 22:04:00

Software Options

100optical	Ethernet 100Base-FX (Optical) Interface	Disabled
GigE_Optical	Ethernet 1000Base-X (Optical) Interface	Enabled
GigE_Electrical	Ethernet 1000Base-T (Electrical) Interface	Enabled
FC-1X	Fibre Channel 1X	Disabled
FC-2X	Fibre Channel 2X	Disabled
TCP-THPUT	Tcp Throughput Test Application	Disabled
FC-4X	Fibre Channel 4X	Disabled
MPLS	MPLS Encapsulation (10GE and less)	Disabled
IPV6	Internet Protocol Version 6 (IPv6) (10GE and less)	Enabled
ETH-THRU	Through Mode Test Application	Enabled
Cable_Test	Cable Test	Disabled
TRAFFIC_GEN	Traffic Generation and Monitoring Test Application	Enabled
CPRI-1.2G	CPRI 1.2288 Gbit/s	Disabled
CPRI	CPRI 2.4576 Gbit/s and 3.072 Gbit/s	Disabled
CPRI-4.9G	CPRI 4.9152 Gbit/s	Disabled
CPRI-6.1G	CPRI 6.144 Gbit/s	Disabled
OBSAI	OBSAI 3.072 Gbit/s	Disabled
1588PTP	1588 Precision Time Protocol	Enabled
SyncE	Synchronous Ethernet	Disabled
ETH-OAM	Carrier Ethernet OAM Test Application	Disabled
ETH-CAPTURE	Ethernet Frame Capture for Advanced Troubleshooting	Disabled
ADV-FILTERS	Advanced Filtering	Disabled
RFC6349	RFC 6349 Test Application (10GE and less)	Disabled
iSAM	Intelligent Service Activation Methodology	Disabled
LINK-OAM	Link OAM	Disabled
TRAFFIC-SCAN	Traffic Scan	Disabled

LOGGER

ID	Time	Event	Duration	Details
1	11/11 12:56:12	Test Started		2016-11-11
2	11/11 13:56:14	Test Stopped		PASS