HB0825

Handbook

CoreQSGMII V2.0

04 2018

Microsemi
Power Matters.™

About Microsemi

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions; security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com.

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
E-mail: sales.support@microsemi.com
www.microsemi.com

Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Release 1.0

Revision 1.0 is the first publication of this document. Created for CoreQSGMII v2.0.

Contents

List of Figures

List of Tables

2 Introduction

2.1 Overview

The CoreQSGMII provides a solution to combine four SGMII lines into a single 5.0 Gbps link. The CoreQSGMII module takes each of the four gigabit/media independent interface data stream and encodes it into 10-bit symbols per port to form a 40-bit interface to the Transceiver. In the Receive direction, the 40-bit symbols are decoded and converted into the receive G/MII signal set. The CoreQSGMII module is managed and monitored through the management data input/output (MDIO) interface.

The CoreQSGMII supports Auto-Negotiation for each of the four ports, which allows two link partners to exchange details of capabilities and hence determine the appropriate link operation characteristics, including duplex mode and flow control direction.

2.2 Features

- Compliant with Cisco Systems' proprietary QSGMII Specification, Revision 1.2
- Four independent 10/100/1000 Mbps ports
- Implement SGMII adaptation for 10/100/1000 operation of each port
- Implements 8b/10b encoder/decoder and PCS Transmit function for each port in accordance with IEEE Standard 802.3 Clause 36 modified to support QSGMII
- K28.5 swapper on Port 0 in the transmit path as per the QSGMII specification
- Comma alignment and K28.1 swapper on Port 0 in the receive path as per the QSGMII specification
- Supports auto-negotiation functionality per port
- Supports management interface for register configuration through MDIO interface
- The SerDes interface on the QSGMII IP is configured for 40-bit, single lane operating at 5.0 Gbps speed with fabric interface running at 125MHz

2.3 Core Version

This Handbook applies to CoreQSGMII version 2.0.

2.4 Supported Families

This version of CoreQSGMII supports the following families:

- PolarFire®

2.5 Device Utilization and Performance

Utilization and performance data is listed in Table 1 for the supported device families. The data listed in this table is indicative only. The overall device utilization and performance of the core is system dependent.

Table 1 Device Utilization and Performance

Family	Titles			Utilization		Performance (MHz)
	Sequential	Combinatorial	Total	Device	Total %	
PolarFire	4476	9602	14078	MPF300T	4.6	160

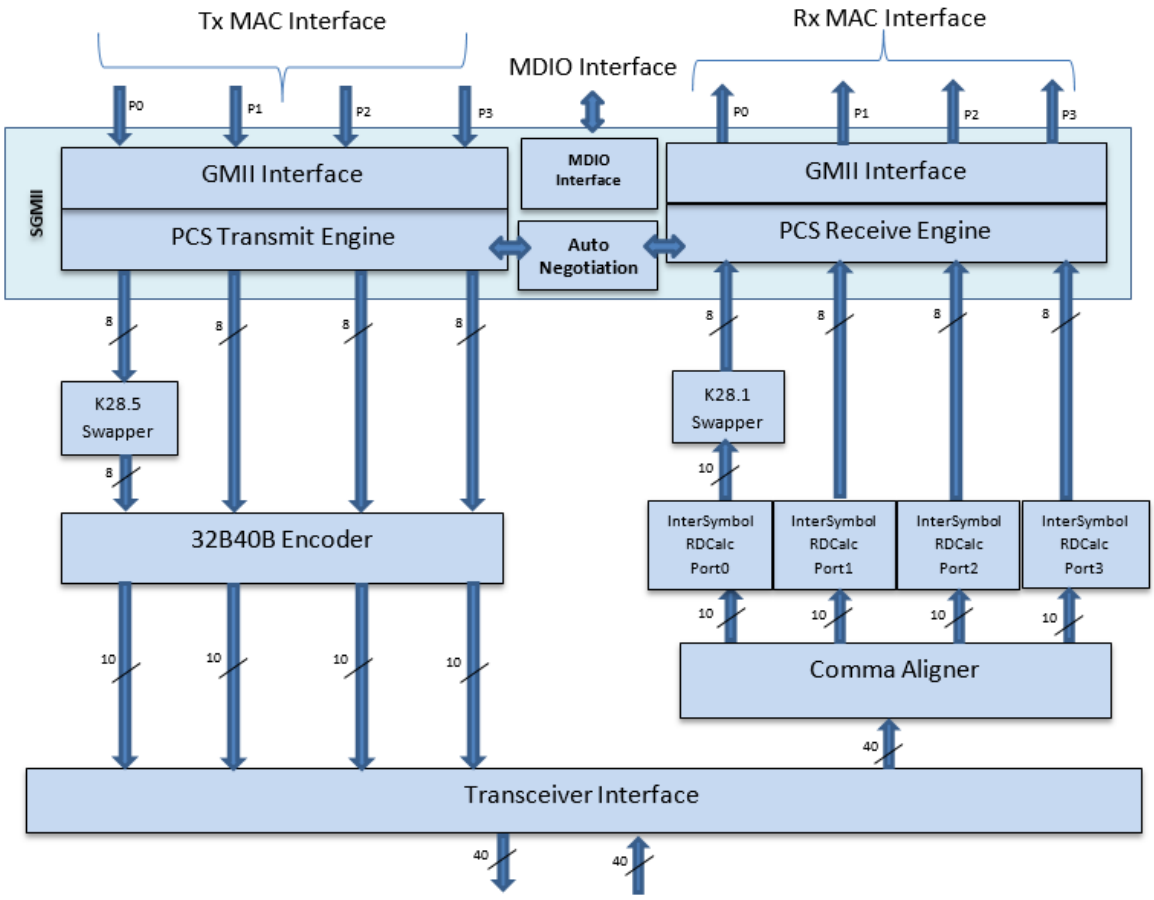
Notes: The data in this table is achieved using typical synthesis and layout settings. Frequency (in MHz) was set to 125 and speed grade was -1.

3 Functional Description

3.1 CoreQSGMII Interfaces

Figure 1 shows the CoreQSGMII block diagram.

Figure 1 CoreQSGMII Block Diagram



The IP provides four independent ports of 1000/100/10 Mbps for Ethernet communication between the PHY and the MAC over a single link of 5.0 Gbps using transceiver configured in 40-bit mode.

The IP contains PCS functionality logic modified in compliance with Cisco's QSGMII Specification Revision 1.2.

CoreQSGMII consists of the following interfaces:

- MAC-side (G/MII) Interface
- Management Interface
- Transceiver Interface

3.1.1 MAC-side (G/MII) Interface

The IP connects to four MACs through four independent IEEE 802.3 gigabit media independent G/MII Interface.

Gigabit media-independent Interface (G/MII) is an interface between the media access control (MAC) device and physical layer (PHY). It defines speeds up to 1000 Mbps, implemented using an8-bit data interface clocked at 125 MHz, and is backwards compatible with the media-independent interface (MII) specification. It can also operate at speeds of 10 or 100 Mbps as per the MII specification.

Data on the interface is framed using the IEEE Ethernet standard. It consists of the following:

- Preamble
- Start frame delimiter
- Ethernet headers
- Protocol specific data
- Cyclic redundancy check (CRC)

In case of G/MII transmission, there are two clocks, depending on whether the PHY is operating at 1000 Mbps or 10/100 Mbps speeds. TBI_TX_CLK is supplied to the PHY for 1000 Mbps speed, and the transmit data and control signals are synchronized to this. Otherwise, for 10/100 Mbps, the TXCLK supplied by the PHY is used for synchronizing those signals. This operates at either 25 MHz for 100 Mbps, or 2.5 MHz for 10 Mbps connections. The RXCLK is 2.5/25/125 MHz for 10/100/1000 respectively, is supplied from PHY/Transceiver.

It contains the logic for PCS transmit path data conversion (repeating each data byte 10/100 times for 10/100 IEEE 802.3 gigabit media independent G/MII Interface Mbps) and synchronization with the TBI_TX_CLK. For Port 0, the PCS transmit logic transmits only /I1/ ordered sets instead of /I2/ per the QSGMII specification. For other ports, the PCS transmit logic transmits both /I1/ and /I2/ ordered sets.

It also performs the receive path data conversion from ten-bit interface to G/MII data on RXCLK domain. It stores the data every 10/100 clocks for 10/100 Mbps respectively.

3.1.2 Management Interface

The CoreQSGMII registers are accessed through the MDIO interface. A Single MDIO interface is provided for the four independent ports. The MDIO controller in the MAC can read and write the control and status registers of the CoreQSGMII. The MDIO interface is compliant with the IEEE 802.3z Clause 45.

3.1.3 Transceiver Interface

The 10-bit data from each of the SGMII instances is transferred to the transceiver interface forming 40-bit data. In the receive path, the 40-bit data from the transceiver interface is transferred to Comma alignment and K28.1 detection logic.

The transceiver is configured to operate on single lane, 40-bit mode with 125MHz speed at the fabric interface to give 5.0 Gbps link rate.

3.2 Functional Blocks

CoreQSGMII contains the following blocks:

- Four SGMII blocks
- Tx / Rx Swapper
- 32B40B Encoder
- Comma Aligner
- Receive Intersymbol RD Swapper

3.2.1 SGMII Block

Each SGMII block has the G/MII interface to connect the MAC with the PHY. It performs the transmit exchange functionality, receive exchange functionality, auto-negotiation and the management functionality using the MDIO interface.

TEX (Transmit Exchange functionality)

This module performs clause 36 transmit related functionality of 802.3z.

The PCS transmit functionality is modified as per QSGMII specification Revision 1.2 (Figure 1) on Port 0 by detaching the PCS transmit functionality from the 8b/10b encoder. This is done by incorporating a “K28.5” swapper function that modifies the IDLE /I/ and Configuration /C/ ordered sets by replacing every occurrence of /K28.5/ with /K28.1/. The swapper function operates on the GMII octets before the 8b/10b encoding. It is important to note that the transmitter generates only /I1/ ordered set and not /I2/ as per the QSGMII specification.

TEX operation is governed by auto negotiation, which provides CFG/IDL/DAT information.

In CFG mode, TEX sends /C/ ordered sets with data from ANX.

In IDL mode, TEX sends /I/ order sets.

In DAT mode, TEX send 8b10b encoded packets.

REX (Receive Exchange functionality)

Performs clause 36 receive related functionality of 802.3z.

Performs comma alignment and passes aligned two-code-group. It determines the alignment by searching for the comma character K28.1 on the received 40-bit data, as the Port 0 data can be on any of the four 10-bit lanes. After a match is found, the K28.1 swapper logic swaps the K28.1 code-group received on the port 0 with K28.5 code-group as per the QSGMII specification.

The code-groups from PCS are decoded for 10b8b and inspected by the receive logic.

The received 10-bit code-groups from the Ten-Bit Interface performs exact reverse procedure as that of the transmit function by undoing the swapping function introduced in the transmit path. That is, a “K28.1” swapper function replaces the received /K28.1/ with /K28.5/ for every occurrence on the Port 0. The swapper logic works post the 10b/8b decoding on the 8-bit GMII octets. However, the carrier detect function operates on the 10B code-group to detect /K28.1/ for Port0.

In Auto-Negotiation mode as indicated by the Transmit variable, the PCS module looks for Configuration ordered sets and passes the Receive Configuration Register contents to the ANX module.

After Auto-Negotiation completes, removes the encapsulation codes and passes the received packet.

3.2.2 K28.5 Swapper

The Port 0 transmit side incorporates a K28.5 swapper logic which modifies each occurrence of K28.5 with K28.1. This is done in order to determine the port number based data alignment. The transmit data will appear on the QSGMII link in the order of port0 first, then port1, then port2 and then port3. This is repeated with port 0 data again on the link and so on.

Note: The ports other than port0 does not perform K28.5 swapping.

Table 2 Port0 "K28.5" Swapper Definition

Code	Ordered_Set	Number of Code Groups	Port 0 "pre-swapper" Encoding	Port 0 "post-swapper" Encoding
/C/	Configuration		Alternating /C1/ and /C2/	Alternating /C1/ and /C2/
/C1/	Configuration 1	4	/K28.5/D21.5/Config_Reg	/K28.1/D21.5/Config_Reg
/C2/	Configuration 2	4	/K28.5/D2.2/Config_Reg	/K28.1/D2.2/Config_Reg
/I/	IDLE		Correcting /I1/	Correcting /I1/
/I1/	IDLE 1	2	/K28.5/D5.6/	/K28.1/D5.6/

3.2.3 32B40B Encoder

The 8-bit data from each of the SGMII blocks is formed into a 32-bit data. This module performs the 32B40B PCS encoding. The running disparity between the transmitted 10-bit symbols from each SGMII block is handled in this block.

3.2.4 Comma Aligner Block

The Comma Aligner block performs alignment on two consecutive 40-bit data received from the transceiver interface. The Comma Aligner block performs barrel shifting on the incoming data and searches for K28.1 Comma code. The detection of Comma code determines the 10-bit symbol alignment with respect to Port 0 data.

It provides the 40-bit aligned data with each 10-bit data going to each of the four InterSymbol RD swapper modules.

3.2.5 InterSymbol RD Swapper

This block calculates the running disparity between the successive 10-bit aligned data in the receive path. The output is 10-bit data with the running disparity maintained between successive 10-bit symbols.

3.2.6 K28.1 Swapper Logic

The K28.1 Swapper module receives 10-bit code group from the InterSymbol RD swapper module. It undoes the modification done in the transmit path by replacing back every occurrence of K28.1 with K28.5 symbol code. Every occurrence of K28.1 in IDLE /I/ and configuration /C/ ordered sets is replaced with K28.5.

Table 3 Port0 "K28.1" Swapper Definition

Code	Ordered_Set	Number of Code Groups	Port 0 "pre-swapper" Encoding	Port 0 "post-swapper" Encoding
/C/	Configuration		Alternating /C1/ and /C2/	Alternating /C1/ and /C2/
/C1/	Configuration 1	4	/K28.1/D21.5/Config_Reg	/K28.5/D21.5/Config_Reg
/C2/	Configuration 2	4	/K28.1/D2.2/Config_Reg	/K28.5/D2.2/Config_Reg
/I/	IDLE		Correcting /I1/	Correcting /I1/
/I1/	IDLE 1	2	/K28.1/D5.6/	/K28.5/D5.6/

4 Interface

4.1 Configuration Parameters

CoreQSGMII has parameters (Verilog) for configuring the RTL code, described in Table 4. All parameters are integer types.

Table 4 CoreQSGMII Parameters and Generics Descriptions

Parameter	Valid Range	Default	Description
FAMILY	26	26	Must be set to the required FPGA family: 26 - Polarfire device
MDIO_PHYID_PORT0	0-31	0	MDIO PHY address for Port0
MDIO_PHYID_PORT1	0-31	1	MDIO PHY address for Port1
MDIO_PHYID_PORT2	0-31	2	MDIO PHY address for Port2
MDIO_PHYID_PORT3	0-31	3	MDIO PHY address for Port3

4.2 I/O Signals

The port signals for the CoreQSGMII IP are shown in Table 5.

Table 5 CoreQSGMII I/O Signals

Name	Width	Direction	Description
Clocks and Resets			
TXCLK_P0	1	In	2.5/25/125 MHz transmit clock generated from Transceiver TX clock according to 10/100/1000 Mbps support for port 0.
RXCLK_P0	1	In	2.5/25/125 MHz receive clock generated from Transceiver RX clock according to 10/100/1000 Mbps support for port 0.
TXCLK_P1	1	In	2.5/25/125 MHz transmit clock generated from Transceiver TX clock according to 10/100/1000 Mbps support for port 1.
RXCLK_P1	1	In	2.5/25/125 MHz receive clock generated from Transceiver RX clock according to 10/100/1000 Mbps support for port 1.
TXCLK_P2	1	In	2.5/25/125 MHz transmit clock generated from Transceiver TX clock according to 10/100/1000 Mbps support for port 2.
RXCLK_P2	1	In	2.5/25/125 MHz receive clock generated from Transceiver RX clock according to 10/100/1000 Mbps support for port 2.
TXCLK_P3	1	In	2.5/25/125 MHz transmit clock generated from Transceiver TX clock according to 10/100/1000 Mbps support for port 3.
RXCLK_P3	1	In	2.5/25/125 MHz receive clock generated from Transceiver RX clock according to 10/100/1000 Mbps support for port 3.
TBI_TX_CLK	1	In	125 MHz TBI transmit clock from Transceiver
TBI_RX_CLK	1	In	125 MHz TBI receive clock from Transceiver
RESET	1	In	Asynchronous active high reset. The reset is internally synchronized with the respective clock domains.

G/MII Interface (Port 0)			
TXD_0	8	In	G/MII transmit data
TXEN_0	1	In	G/MII transmit enable
TXER_0	1	In	G/MII transmit error
RXD_0	8	Out	G/MII Receive data
RXDV_0	1	Out	G/MII Receive data valid
RXER_0	1	Out	G/MII Receive error
COL_0	1	Out	MII collision
CRS_0	1	Out	MII carrier sense
G/MII Interface (Port 1)			
TXD_1	8	In	G/MII transmit data
TXEN_1	1	In	G/MII transmit enable
TXER_1	1	In	G/MII transmit error
RXD_1	8	Out	G/MII Receive data
RXDV_1	1	Out	G/MII Receive data valid
RXER_1	1	Out	G/MII Receive error
COL_1	1	Out	MII collision
CRS_1	1	Out	MII carrier sense
G/MII Interface (Port 2)			
TXD_2	8	In	G/MII transmit data
TXEN_2	1	In	G/MII transmit enable
TXER_2	1	In	G/MII transmit error
RXD_2	8	Out	G/MII Receive data
RXDV_2	1	Out	G/MII Receive data valid
RXER_2	1	Out	G/MII Receive error
COL_2	1	Out	MII collision
CRS_2	1	Out	MII carrier sense
G/MII Interface (Port 3)			
TXD_3	8	In	G/MII transmit data
TXEN_3	1	In	G/MII transmit enable
TXER_3	1	In	G/MII transmit error
RXD_3	8	Out	G/MII Receive data
RXDV_3	1	Out	G/MII Receive data valid
RXER_3	1	Out	G/MII Receive error
COL_3	1	Out	MII collision
CRS_3	1	Out	MII carrier sense

Ten-Bit Interface (Port 0)			
RCG_0	10	In	TBI Receive code group for port 0
TCG_0	10	Out	TBI Transmit code group for port 0
Ten-Bit Interface (Port 1)			
RCG_1	10	In	TBI Receive code group for port 1
TCG_1	10	Out	TBI Transmit code group for port 1
Ten-Bit Interface (Port 2)			
RCG_2	10	In	TBI Receive code group for port 2
TCG_2	10	Out	TBI Transmit code group for port 2
Ten-Bit Interface (Port 3)			
RCG_3	10	In	TBI Receive code group for port 3
TCG_3	10	Out	TBI Transmit code group for port 3
Other Ten-Bit Interface Signals			
TBI_RX_READY	1	In	RCG valid, recommended to connect with Transceiver Ready
TBI_TX_VALID	1	Out	TCG valid, recommended to connect with Transceiver transmit valid Note: This signal is not available for PolarFire
TBI_RX_VALID	1	In	Available for PolarFire, recommended to connect with Transceiver receive valid
MDIO Interface			
MDC	1	In	Management data clock, recommended to drive 2.5 MHz
MDO	1	In	Management data output
MDOEN	1	In	Management data output enable
MDI_EXT	1	In	Management data input from external PCS/PHY
MDI	1	Out	MII Management data Input

5 Register Map and Descriptions

The following registers are accessed through the MDIO interface clause 22 of the IEEE 802.3 specification. The PHY address for the MDIO registers can be configured.

The CoreQSGMII contains the management registers specified in IEEE 802.3, Clause 37 – Control, Status, Auto Negotiation Advertisement, Link Partner Ability, Auto Negotiation Expansion and Extended Status. The register set is read/write through MDIO interface.

Table 6 CoreQSGMII Register Map

ADR	Register
00h	Control
01h	Status
04h	AN Advertisement
05h	AN Link Partner Base Page Ability
06h	AN Expansion
07h	AN Next Page Transmit
08h	AN Link Partner Ability Next Page
0Fh	Extended Status
10h	Jitter Diagnostics
11h	TBI Control

Table 7 Register Descriptions

Address	Function
0x00	**Control** **[15] (R/W, SC) PHY RESET: Default 0** Setting this bit causes the TEX, REX, and ANX sub-modules in the CoreSGMII core to be reset. This bit is self-clearing. **[14] (R/W) LOOP BACK: Default 0** Setting this bit causes the transmit output of the CoreSGMII to be connected to the receive inputs. Clearing this bit result in normal operation. **[13] Reserved.** **[12] (R/W) AUTO-NEGOTIATION ENABLE Default 0** Setting this bit enables the Auto-negotiation process. **[11:10] Reserved.** **[9] (R/W, SC) RESTART AUTO-NEGOTIATION: Default 0** Setting this bit causes the Auto-negotiation process to restart. This action is only available when Auto-Negotiation has been enabled. **[8:0] Reserved.**
0x01	**Status - Default Value 0x149** **[15:9] Reserved.** **[8] (RO) EXTENDED STATUS: Default 1** This bit indicates that PHY status information. **[7] Reserved.**

	[6] (RO) MF PREMABLE SUPPRESSION ENABLE: Default 1 This bit indicates whether the PHY is capable of handling MII Management Frames without the 32-bit preamble field. Returns 1 to indicate support for suppressing preamble MII Management Frames. **[5] (RO) AUTO-NEGOTIATION COMPLETE:** When 1, this bit indicates that the Auto-negotiation process has completed. This bit returns '0' when either the Auto-negotiation process is underway or the Auto-negotiation function is disabled. **[4] (RO) REMOTE FAULT: Default 0** When 1, a remote fault condition has been detected between the CoreSGMII and the PHY. **[3] (RO) AUTO-NEGOTIATION ABILITY: Default 1** When 1, this bit indicates that the CoreSGMII has the ability to perform Auto-negotiation. **[2] (RO) LINK STATUS: Default 0** When 1, this bit indicates that a valid link has been established between the CoreSGMII and PHY. When 0, no valid link has been established. **[1] Reserved.** **[0] (RO) EXTENDED CAPABILITY: Default 1** This bit indicates that the CoreSGMII contains the extended set of registers.
0x02	**Reserved**
0x03	**Reserved**

<table>
<tr><th>Address</th><th>Function</th></tr>
<tr><td>0x04</td><td>
AN Advertisement (SGMII)
<table>
<tr><th>Bit</th><th>Tx_config from PHY to MAC</th><th>Tx_config from MAC to PHY</th></tr>
<tr><td>15</td><td>Link: 1=link up, 0=link down</td><td>0:Reserved</td></tr>
<tr><td>14</td><td>Reserved for AN ACK</td><td>1</td></tr>
<tr><td>13</td><td>0:Reserved</td><td>0:Reserved</td></tr>
<tr><td>12</td><td>Duplex mode: 1=full, 0=half</td><td>0:Reserved</td></tr>
<tr><td>11:10</td><td>Speed:
11 = Reserved 10 = 1000 Mbps
01 =100 Mbps 00 = 10 Mbps</td><td></td></tr>
<tr><td>9:1</td><td>0:Reserved</td><td>0:Reserved</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
</table>
[15] (R/W) LINK UP: Assertion of this bit indicates that the link is up. When the CoreSGMII integrated within MAC this bit must be written ‘0’.
[14] (RO) ACK (Reserved). Ignore on read.
[13] (R/W) RESERVED:
This bit must be written ‘0’ for correct CoreSGMII operation.
[12] (R/W) FULL-DUPLEX:
The assertion of this bit indicates that the link is transferring data in full duplex mode. When the CoreSGMII is integrated within MAC this bit must be written ‘0’.
[11:10] (R/W) LINK SPEED:
Link speed set by the application for the Auto negotiation. These bits have no relation with the SPEEDO port signal.
[9:0] (R/W): These bits must always be written ‘0000000001’ for correct CoreSGMII operation.
</td></tr>
</table>

<table>
<tr><th>Address</th><th>Function</th></tr>
<tr><td>0x04</td><td>
AN Advertisement (1000BASE-X)
[15] (R/W) NEXT PAGE: Default 0
The local device asserts this bit to either request Next Page transmission or advertise Next Page exchange capability. This bit can thus be set when the local has no Next Pages but wishes to allow reception of Next Pages. If the local device has no Next Pages, and the Link Partner wishes to send Next Pages, the local device must send Null Message Codes and have the MESSAGE PAGE set to 0b000_0000_0001. This bit must be cleared where the local device wishes not to engage in Next Page exchange.
[14] Reserved.
[13:12] (R/W) REMOTE FAULT: Default 0
Encodes the local device’s remote fault condition. A fault may be indicated by setting a non-zero Remote Fault encoding and re-negotiating.
<table>
<tr><th>RF1 (4.12)</th><th>RF2 (4.13)</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>No error, link Ok.</td></tr>
<tr><td>0</td><td>1</td><td>Offline.</td></tr>
<tr><td>1</td><td>0</td><td>Link_Failure</td></tr>
<tr><td>1</td><td>1</td><td>Auto-Negotiation_Error.</td></tr>
</table>
[11:9] Reserved
[8:7] (R/W) PAUSE: Encodes the local device’s PAUSE capability.
Pause Encoding:
<table>
<tr><th>PAUSE1 (4.7)</th><th>ASM_DIR (4.8)</th><th>Capability</th></tr>
<tr><td>0</td><td>0</td><td>No PAUSE.</td></tr>
<tr><td>0</td><td>1</td><td>Asymmetric PAUSE toward link partner.</td></tr>
<tr><td>1</td><td>0</td><td>Symmetric PAUSE.</td></tr>
<tr><td>1</td><td>1</td><td>Asymmetric PAUSE toward local device.</td></tr>
</table>
[6] (R/W) HALF-DUPLEX:
Setting this bit means the local device is capable of half-duplex operation.
[5] (R/W) FULL-DUPLEX:
Setting this bit means the local device is capable of full-duplex operation.
[4:0] Reserved
</td></tr>
<tr><td>0x5</td><td>
AN Link Partner Base Page Ability (SGMII)
[15] (RO) LINK UP: This bit indicates that the link is up.
[14:13] Reserved
[12] (RO) FULL DUPLEX: Link partner full duplex ability received.
[11:10] (RO) LINK SPEED: Link partner speed ability received.
[9:0] (R/W): These bits must always be written 0000000001 for correct CoreSGMII operation.
</td></tr>
</table>

<table>
<tr><th>Address</th><th>Function</th></tr>
<tr><td>0x5</td><td>
AN Link Partner Base Page Ability (1000BASE-X)
[15] (RO) NEXT PAGE:
The Link Partner asserts this bit either to request Next Page transmission or to indicate the capability to receive Next Pages. When 0, the Link Partner has no subsequent Next Pages or is not capable of receiving Next Pages.
[14] (RO) ACK (Reserved): Ignore on read.
[13:12] (RO) REMOTE FAULT: Encodes the Link Partner’s remote fault condition.
<table>
<tr><th>RF1 (4.12)</th><th>RF2 (4.13)</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>No error, link Ok.</td></tr>
<tr><td>0</td><td>1</td><td>Offline.</td></tr>
<tr><td>1</td><td>0</td><td>Link_Failure</td></tr>
<tr><td>1</td><td>1</td><td>Auto-Negotiation_Error.</td></tr>
</table>
[11:9] Reserved
[8:7] (RO) PAUSE: Encodes of the Link Partner’s PAUSE capability.
Pause Encoding:
<table>
<tr><th>PAUSE1 (4.7)</th><th>ASM_DIR (4.8)</th><th>Capability</th></tr>
<tr><td>0</td><td>0</td><td>No PAUSE.</td></tr>
<tr><td>0</td><td>1</td><td>Asymmetric PAUSE toward link partner.</td></tr>
<tr><td>1</td><td>0</td><td>Symmetric PAUSE.</td></tr>
<tr><td>1</td><td>1</td><td>Asymmetric PAUSE toward local device.</td></tr>
</table>
[6] (RO) HALF-DUPLEX: When 1, Link Partner is capable of half-duplex operation. When 0, Link Partner is incapable of half-duplex mode.
[5] (RO) FULL-DUPLEX: When 1, Link Partner is capable of full-duplex operation. When 0, Link Partner is incapable of full-duplex mode.
[4:0] Reserved
</td></tr>
<tr><td>0x06</td><td>
AN Expansion (SGMII)
[15:3] Reserved
[2] (RO) NEXT PAGE ABLE: Default 1
When 1, indicates that the local device supports the Next Page function.
[1] (RO) PAGE RECEIVED:
When 1, indicates that a new page has been received and stored in the applicable AN LINK PARTNER ABILITY or AN NEXT PAGE.
[0] Reserved
AN Expansion (1000BASE-X)
[15:3] Reserved
[2] (RO) NEXT PAGE ABLE: 1 indicates the local device supports Next Page function. Returns 1 on read.
[1] (RO,LH) PAGE RECEIVED:
1 indicates that a new page has been received and stored in the applicable AN LINK PARTNER ABILITY or AN NEXT PAGE register. This bit latches high in order for the software to detect when polling. The bit is cleared on a read to the register.
</td></tr>
</table>

Address	Function
0x07	**AN Next Page Transmit (SGMII)** Use of this register is user dependent. User can define the functionality of bits of this register as per system required. **[15:0] User defined Register** **AN Next Page Transmit (1000BASE-X)** **[15] (R/W) NEXT PAGE:** Assert this bit to indicate additional Next Pages to follow. The bit is cleared to indicate last page. **[14] (RO) ACK (Reserved): Write '0', ignore on read.** **[13] (R/W) MESSAGE PAGE:** Assert bit to indicate Message Page. Clear bit to indicate Unformatted Page. **[12] (R/W) ACKNOWLEDGE 2:** Used by Next Page function to indicate device has ability to comply with the message. Assert bit if local device complies with message. Clear bit if the local device cannot comply with the message. **[11] (RO) TOGGLE:** Used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes the opposite value of the Toggle bit of the previously exchanged Link Code Word. The initial value in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word. **[10:0] (R/W) MESSAGE / UNFORMATTED CODE FIELD:** Message pages are formatted pages that carry a predefined Message Code, which is enumerated in IEEE 802.3u/Annex 28C. Unformatted Code Fields take on an arbitrary value.
0x08	**AN Link Partner Ability Next Page (SGMII)** Use of this register is user dependent. User can define the functionality of bits of this register as per system required. **[15:0] User defined Register** **AN Link Partner Ability Next Page (1000BASE-X)** **[15] (RO) NEXT PAGE:** The Link Partner asserts this bit to indicate additional Next Pages to follow. When 0, indicates last Next Page from link partner. **[14] (RO) ACK (Reserved): Ignore on read.** **[13] (RO) MESSAGE PAGE:** When 1, indicates Message Page. When '0', indicates Unformatted Page. **[12] (RO) ACKNOWLEDGE 2:** Indicates Link Partner's ability to comply with the message. When 1, Link Partner complies with message. When 0, Link Partner cannot comply with the message. **[11] (RO) TOGGLE:** Used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes the opposite value of the Toggle bit of the previously exchanged Link Code Word. The initial value in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word. **[10:0] (RO) MESSAGE / UNFORMATTED CODE FIELD:** Message pages are formatted pages that carry a predefined Message Code, which is enumerated in IEEE 802.3u/Annex 28C. Unformatted Code Fields take on an arbitrary value.

<table>
<tr><th>Address</th><th>Function</th></tr>
<tr><td>0x0F</td><td>
Extended Status
[15] (RO) 1000BASE-X FULL-DUPLEX: Default 1
When 1, indicates PHY can operate in 1000BASE-X full-duplex mode. When ‘0’, indicates PHY cannot operate in this mode.
[14] (RO) 1000BASE-X HALF-DUPLEX: Default 0
When 1, indicates PHY can operate in 1000BASE-X half-duplex mode. When 0, indicates PHY cannot operate in this mode.
[13] (RO) 1000BASE-T FULL-DUPLEX: Default 1
When 1, indicates PHY can operate in 1000BASE-T full-duplex mode. When 0, indicates PHY cannot operate in this mode.
[12] (RO) 1000BASE-T HALF-DUPLEX: Default 0
When 1, indicates PHY can operate in 1000BASE-T half-duplex mode. When 0, indicates PHY cannot operate in this mode.
[11:0] Reserved
</td></tr>
<tr><td>0x10</td><td>
Jitter Diagnostics
[15] (R/W) JITTER DIAGNOSTIC ENABLE: Default 0
Set this bit to enable the CoreSGMII to transmit the jitter test patterns defined in IEEE 802.3z 36A. Clear this bit to enable normal transmit-operation.
[14:12] (R/W) JITTER PATTERN SELECT: Default is 0
Selects the jitter pattern to be transmitted in diagnostics mode.
Jitter Pattern Select Encodings
<table>
<tr><th>Bit 14</th><th>Bit 13</th><th>Bit 12</th><th>Jitter Pattern Select</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>User Defined Custom Pattern</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>802.3z 36A Defined High Frequency 10101010101010101010…</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>802.3z 36A Defined Mixed Frequency 11111010110000010100…</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Custom Defined Low Frequency 11111000001111100000…</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Random Jitter Pattern</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>802.3z 36A Defined Low Frequency 11111000001111100000…</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Reserved</td></tr>
</table>
[11:10] Reserved
[9:0] (R/W) CUSTOM JITTER PATTERN: Default 0
Used in conjunction with JITTER PATTERN SELECT and JITTER DIAGNOSTIC ENABLE. Set this field to the desired custom pattern.
</td></tr>
<tr><td>0x11</td><td>
Ten Bit Interface Control
[15] (R/W) SOFT RESET: Default 0
This bit resets the functional modules in the CoreSGMII. Clear it for normal operation.
[14] (R/W) SHORTCUT LINK TIMER: Default 0
Set this bit 1 to reduce the value of Go Link Timer and Sync. Status Fail Timer to 64 clock pulse. This reduces the simulation time needed to time the 1.6ms Link Timer. Clear it for normal operation. In normal operation, the value of Go Link Timer is 200000 clock pulses and the value of the Sync. Status Fail Timer is 1250000 clock pulses.
[13] (R/W) DISABLE RECEIVE RUNNING DISPARITY: Default 0
</td></tr>
</table>

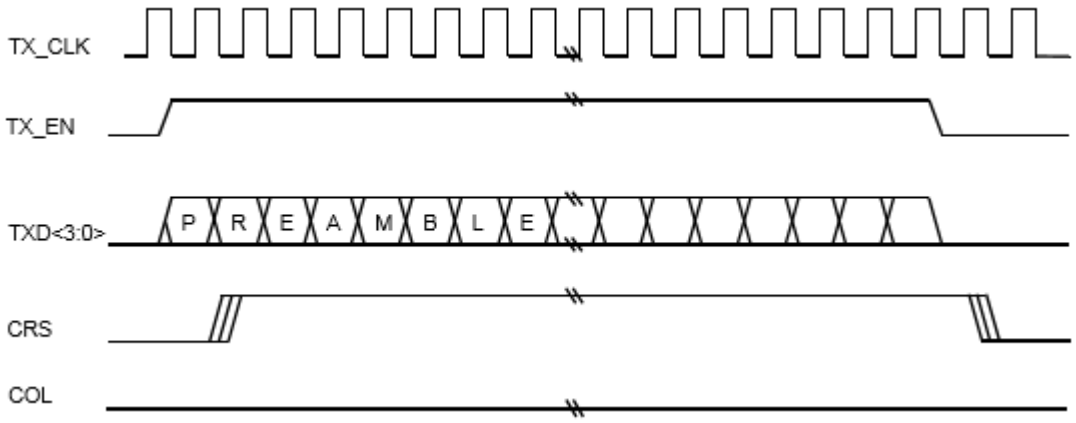
<table>
<tr><th>Address</th><th>Function</th></tr>
<tr><td></td><td>
Set this bit to disable the running disparity calculation and checking in the receive direction. This bit must be 0 for correct CoreSGMII operation.
[12] (R/W) DISABLE TRANSMIT RUNNING DISPARITY: Default 0
Set this bit to disable the running disparity calculation and checking in the transmit direction. This bit must be 0 for correct CoreSGMII operation.
[11] (R/W) GO LINK TIMER VALUE CONTROL: Default 0
When 0 the Go Link Timer Value=1.6ms
When set to 1 the Go Link Timer Value=10ms
[10:9] Reserved
[8] (R/W) AUTO-NEGOTIATION SENSE: Default 0
Set this bit to allow the Auto-Negotiation function to sense either a MAC in Auto-Negotiation bypass mode or an older MAC without Auto-Negotiation capability. When sensed, Auto-Negotiation Complete becomes true; however, Page Received will be low, indicating no page is exchanged. Management can then act accordingly. Clear this bit when IEEE 802.3z Clause 37 behavior is desired, which results in the link not coming up.
[7:4] Reserved
[3:2] (R/W) SPEEDO: Default '10'
<table>
<tr><th>Bit 3</th><th>Bit 2</th><th>Description</th></tr>
<tr><td>1</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>0</td><td>1000 Mbps</td></tr>
<tr><td>0</td><td>1</td><td>100 Mbps</td></tr>
<tr><td>0</td><td>0</td><td>10 Mbps</td></tr>
</table>
[1:0] Reserved
</td></tr>
</table>

6 Timing Diagrams

6.1 G/MII Timing Diagrams

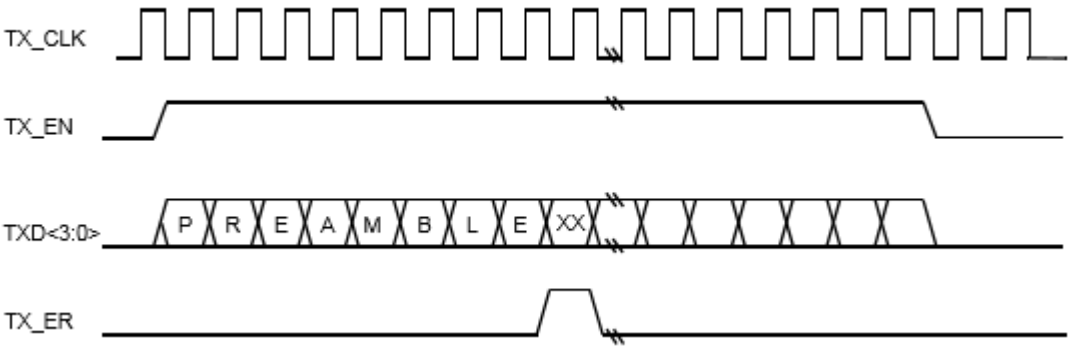
6.1.1 Transmission with no Collision

Figure 2 Transmission with no Collision



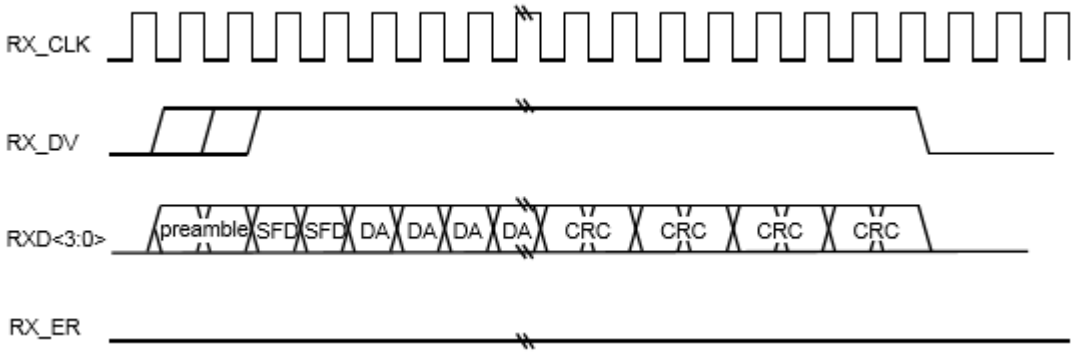
6.1.2 Propagating an Error

Figure 3 Propagating an Error



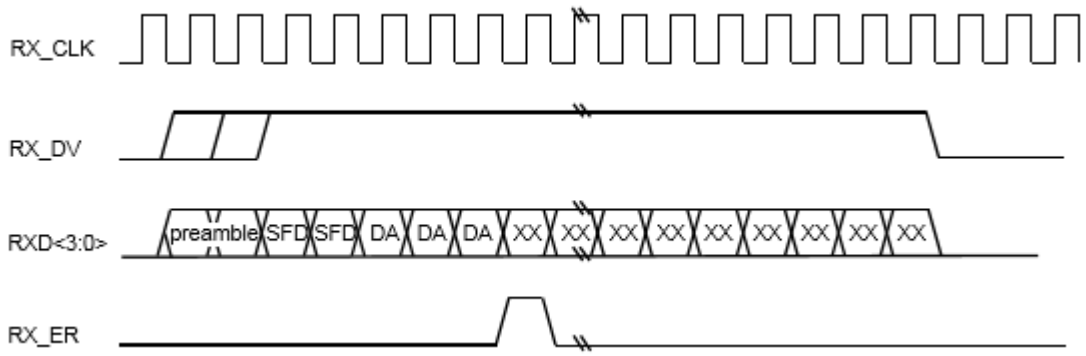
6.1.3 Reception with no Errors

Figure 4 Reception with no Errors



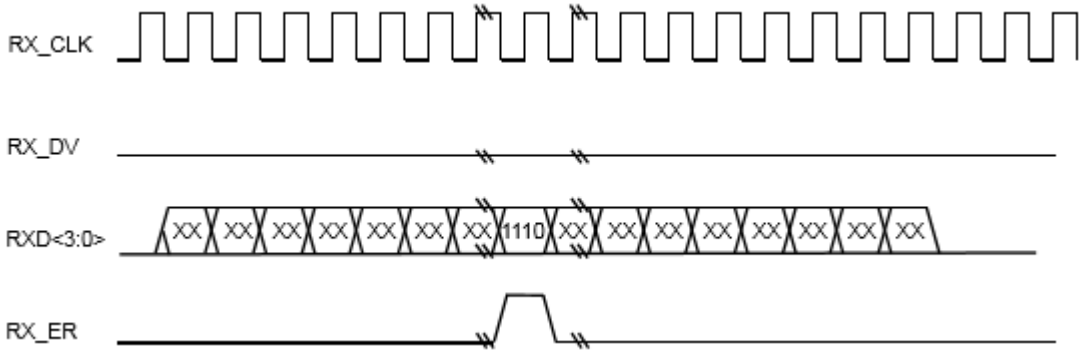
6.1.4 Reception with Errors

Figure 5 Reception with Errors



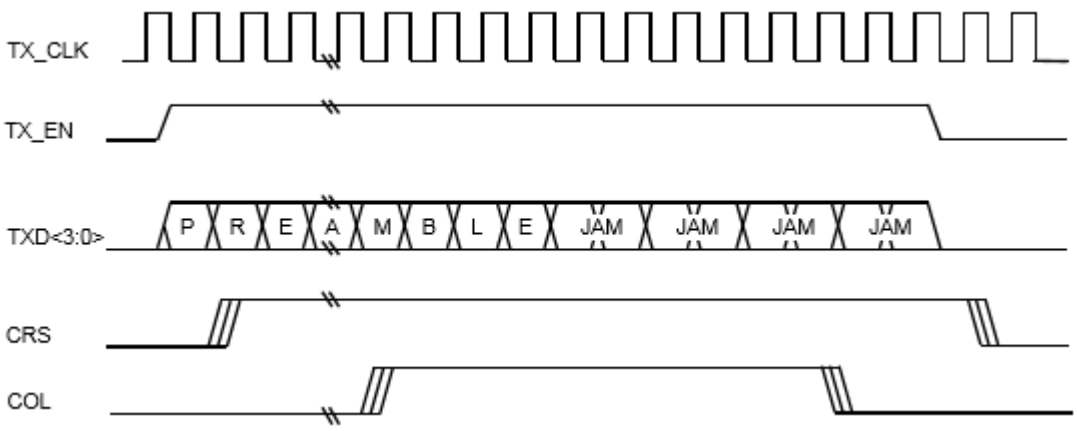
6.1.5 False Carrier Indication

Figure 6 False Carrier Indication



6.1.6 Transmission with Collision

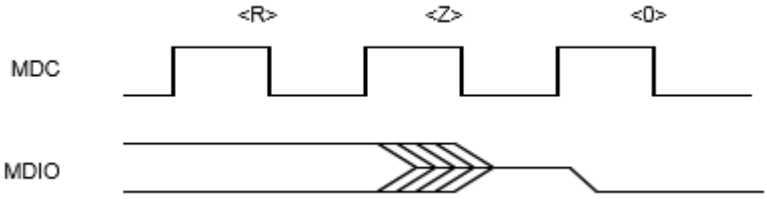
Figure 7 Transmission with Collision



6.2 MDIO Timing Diagrams

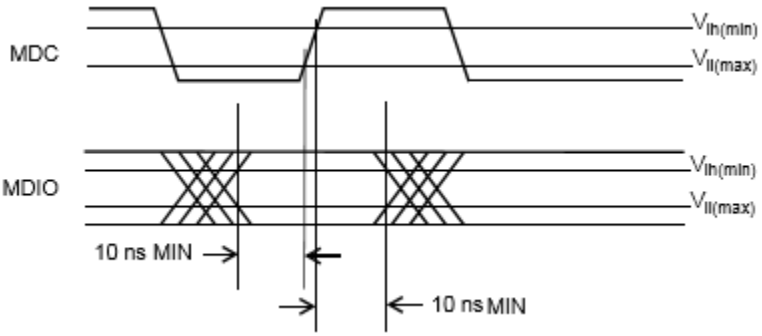
6.2.1 Behavior of MDIO during TA field of a Read Transaction

Figure 8 Behavior of MDIO during TA field of a Read Transaction



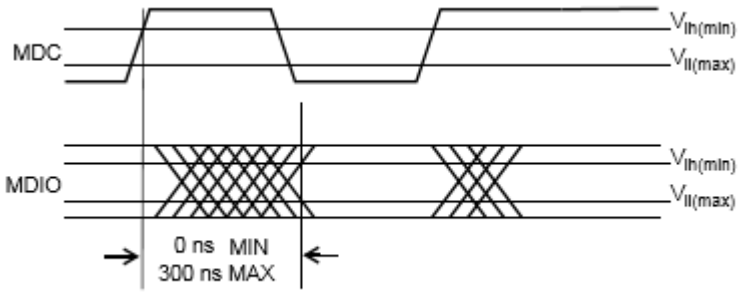
6.2.2 MDIO sourced by STA

Figure 9 MDIO sourced by STA



6.2.3 MDIO sourced by PHY

Figure 10 MDIO sourced by PHY



7 Tool Flow

7.1 License

CoreQSGMII is available with evaluation and obfuscated RTL license.

7.1.1 Obfuscated

Complete RTL code is provided for the core, enabling the core to be instantiated with SmartDesign. Simulation, Synthesis, and Layout can be performed with Libero software. The RTL code for the core is obfuscated using the IP encryption (encryptP1735.pl) solution.

7.1.2 Evaluation

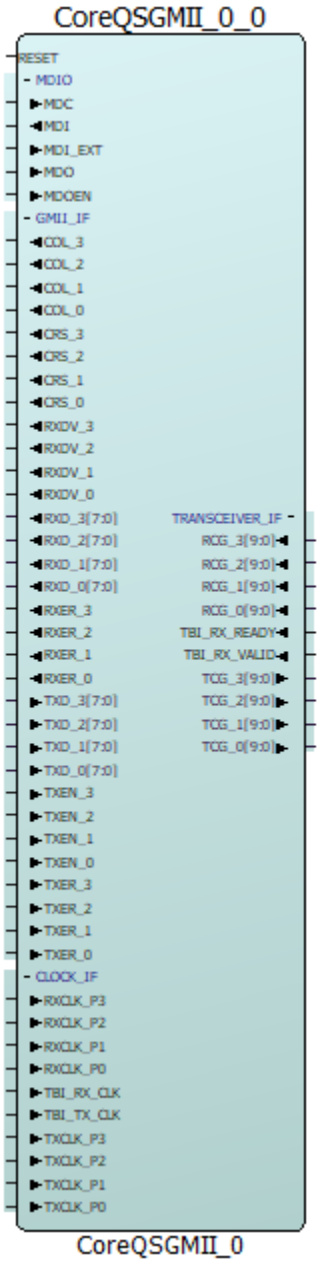
Complete RTL code is provided for the core, enabling the core to be instantiated with SmartDesign. Simulation, Synthesis, and Layout can be performed with Libero software. The RTL code for the core is obfuscated using the IP encryption (encryptP1735.pl) solution and has a self-destruct feature which will stop functioning after 4 hours time at 1.25Gbps data rate.

7.2 SmartDesign

CoreQSGMII is preinstalled in SmartDesign IP Deployment design environment. The core should be configured using the configuration GUI within SmartDesign, as shown in the following figure.

For more information on using SmartDesign to instantiate and generate cores, refer to the Using DirectCore in Libero® SoC User Guide.

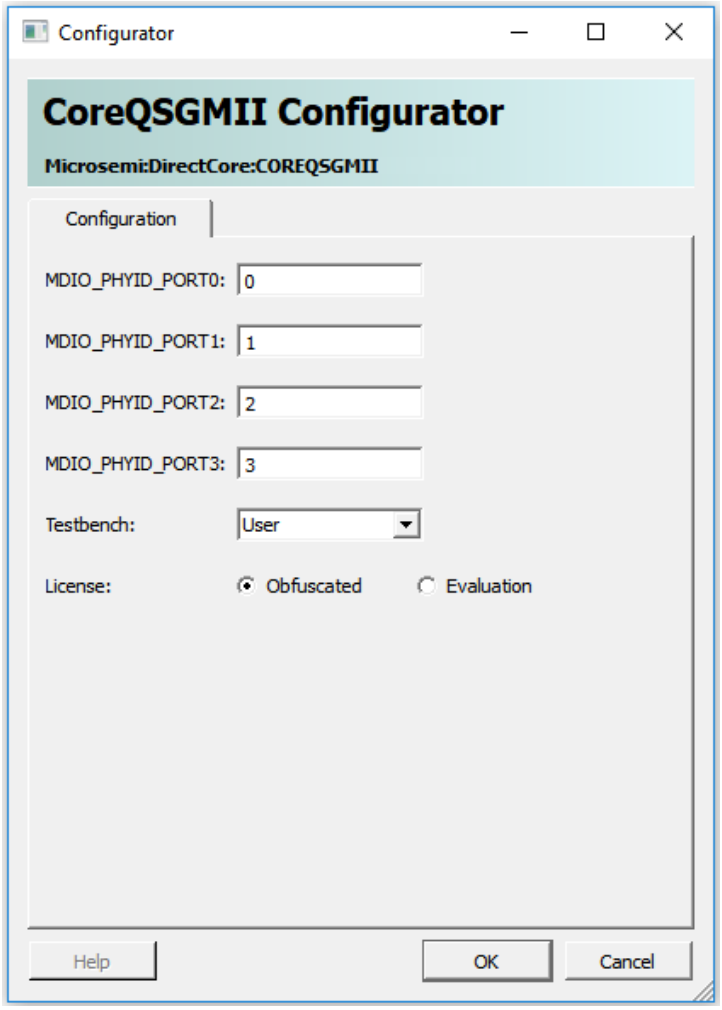
Figure 11 SmartDesign CoreQSGMII Instance View



7.3 Configuring CoreQSGMII in SmartDesign

The following figure shows the options available in Configuration tab.

Figure 12 CoreQSGMII SmartDesign Configuration GUI



7.4 Simulation Flows

The User Testbench for CoreQSGMII is included in all releases.

To run simulations, select the User Testbench flow within the SmartDesign CoreQSGMII configuration GUI, right-click the canvas, and select **Generate Design**.

When SmartDesign generates the design files, it installs the user testbench files.

To run the user testbench, set the design root to the CoreQSGMII instantiation in the Libero SoC design hierarchy pane and click **Simulation** in the **Libero SoC Design Flow** window. This invokes ModelSim® and automatically runs the simulation.

7.5 Synthesis in Libero

After setting the design root appropriately for your design, click **Synthesis** in the Libero SoC software. The **Synthesis** window appears, displaying the Synplicity® project. Set Synplicity to Verilog 2001 standard if Verilog is being used. To run Synthesis, click **Run**.

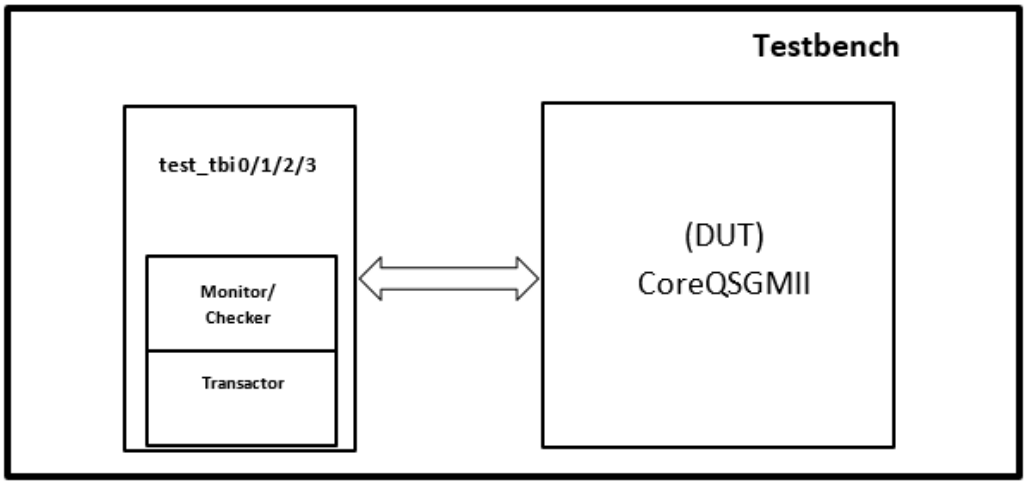
7.6 Place-and-Route in Libero

After the design is synthesized, run the compilation and the place and-route tools. Click the **Layout** icon in the Libero SoC to invoke Designer. CoreQSGMII requires no special place-and-route settings.

8 User Testbench

CoreQSGMII user testbench gives an example of how to use the core.

Figure 13 CoreQSGMII User Testbench

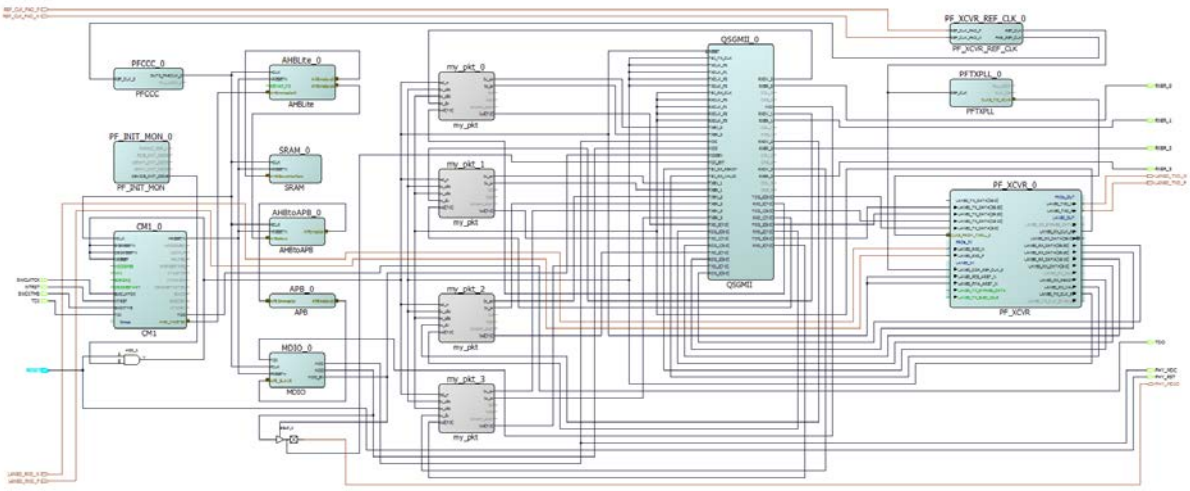


The simulation testbench shown in figure above includes an instantiation of the CoreQSGMII DUT, and Test TBI module. The transactor in the Test TBI module generates the G/ MII data on TXD output and receives RXD input to/ from the DUT (CoreQSGMII). The monitor/ checker checks for the TCG data from the DUT and sends the loopback data to DUT. The purpose of the testbench is to test the functionality of the core by inputting known data, monitoring the output, and checking for expected results.

9 System Integration

The example design explains the CoreQSGMII features and implements the loopback at Transceiver pad level for 1000Mbps mode on each of four ports on the PolarFire Evaluation Kit.

Figure 14 CoreQSGMII Example Design



- On board user RESET and DEVICE_INIT_DONE of PF_INIT_MON_0 are used as reset for QSGMII_0.
- QSGMII_0 has TXCLK_P0, TXCLK_P1, TXCLK_P2, TXCLK_P3, RXCLK_P0, RXCLK_P1, RXCLK_P2, RXCLK_P3, TBI_TX_CLK, TBI_RX_CLK, and MDC clocks.
- TXCLK_P0, TXCLK_P1, TXCLK_P2, TXCLK_P3 and TBI_TX_CLK are connected to 125MHz LANE0_TX_CLK_R of PF_XCVR_0.
- RXCLK_P0, RXCLK_P1, RXCLK_P2, RXCLK_P3 and TBI_RX_CLK are connected to 125MHz LANE0_RX_CLK_R of PF_XCVR_0.
- CoreCortexM1 (CM1_0) application configures the management interface of QSGMII_0 through COREMDIO_APB_0.
- my_pkt_0, my_pkt_1, my_pkt_2 and my_pkt_3 are user defined modules for generating GMII data to each port of QSGMII_0. The TBI data is sent to PF_XCVR_0 and looped back at TX/RX pads of PF_XCVR_0.

Note: Set 'Synthesis gate level netlist format' to 'Verilog netlist' in case this setting is optional in Libero>Project>Project Settings> Design Flow.

Run the Libero flow with enabling the "Timing Driven" and "Repair Minimum Delay Violations" enabled.

Note: The example design can be obtained from the Microsemi technical support team.

9.1 Design Constraints:

```
## Reference clock for PF_XCVR_REF_CLK_0
create_clock -name {REF_CLK_PAD_P} -period 8 [ get_ports { REF_CLK_PAD_P } ]

## TX_CLK clock constraint of PF_XCVR_0
create_clock -name {PF_XCVR_0/LANE0/TX_CLK_R} -period 8 [ get_pins {
PF_XCVR_0/LANE0/TX_CLK_R } ]
## RX_CLK clock constraint of PF_XCVR_0
create_clock -name {PF_XCVR_0/LANE0/RX_CLK_R} -period 8 [ get_pins {
PF_XCVR_0/LANE0/RX_CLK_R } ]
## Generated clock from PFCCC_0
create_generated_clock -name {PFCCC_0/PFCCC_0/pll_inst_0/OUT0} -multiply_by 2 -
divide_by 5 -source [ get_pins { PFCCC_0/PFCCC_0/pll_inst_0/REF_CLK_0 } ] -phase
0 [ get_pins { PFCCC_0/PFCCC_0/pll_inst_0/OUT0 } ]
```

10 Ordering Information

10.1 Ordering Codes

CoreQSGMII can be ordered through your local sales representatives. It must be ordered using the following number scheme: CoreQSGMII-XX, where XX is listed in Table 8.

Note: Evaluation license is also provided with this core.

Table 8·Ordering Codes

XX	Description
OM	Available as obfuscated RTL