HB0439
Handbook
CoreRMII v3.1

Microsemi
a Microchip company

Microsemi 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
Email: sales.support@microsemi.com
www.microsemi.com

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), 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. Learn more at www.microsemi.com.

50200439. 3.0 9/20

Contents

Figures

Tables

1 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 Revision 3.0

Added PolarFire® SoC support.

1.2 Revision 2.0

Updated changes related to CoreRMII v3.0.

1.3 Revision 1.0

The first publication of this document.

2 Introduction

Reduced media independent interface (RMII) is a standard interface, which helps in reducing the number of signals required to connect a PHY to a MAC. CoreRMII is responsible for providing the interface between a standard media independent interface (MII) to RMII conversion. The sixteen-signal MII interface is converted into six-signal RMII interface.

The IP core adheres to RMII specification v1.2. The core has Management Data Input Output (MDIO) interface to transfer the management information between the core and the Station Management Entity (STA) of the MAC. STA controls the link speed, the transfer type and the loopback feature by writing to the control register of the core through the MDIO interface.

2.1 Features

- Provides reduced pin-count interface for Ethernet PHYs.
- Provides media independent interface (MII) towards the microcontroller subsystem (MSS)-side and reduced media independent interface (RMII) on the PHY side.
- Supports 25 MHz clock operation on the MII-side, 50 MHz on the RMII-side for 10/100 Mbps mode operation.
- Provides MDIO interface as defined in clause 22 of IEEE 802.3 standard to control the transfer type, the link speed and the loopback mode.

2.2 Core Version

This Handbook applies to CoreRMII version 3.1.

2.3 Supported Families

- PolarFire® SoC
- PolarFire®
- IGLOO®2
- SmartFusion®2

2.4 Device Utilization and Performance

Utilization and performance data is provided in Table 1 for the SmartFusion2 (M2S050), IGLOO2 (M2GL050), PolarFire (MPF300T_ES) devices families. The data given in Table 1 is indicative only. The overall device utilization and performance of the core is system dependent.

Table 1 • **Device Utilization and Performance**

	Logic Elements				Performance
Family	**DFF**	**4LUT**	**Total**	**%**	**REFCLK (MHz)**
SmartFusion2	170	248	418	0.37	235
IGLOO2	170	248	418	0.37	235
PolarFire	170	247	417	0.07	358

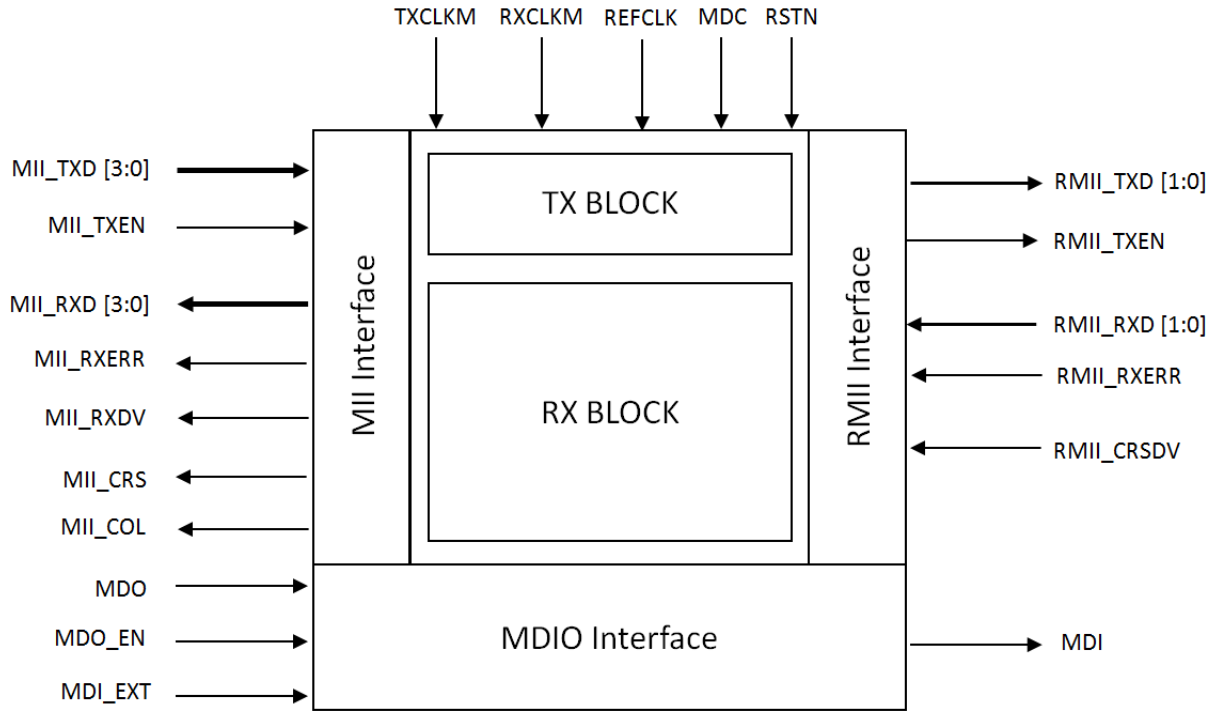
Note: FPGA resources and performance data for the PolarFire SoC family is similar to PolarFire family.

Note: The data in Table 1 was achieved using typical synthesis and layout settings. Frequency (in MHz) was set to 100 and speed grade was -1. The parameter MDIO_PHYID was set to 0.

3 Functional Description

As shown in Figure 1, CoreRMII consists of three major functional blocks: the TX block, the RX block, and the MDIO interface block. CoreRMII has MII interface on MAC-side and RMII interface on PHY-side. The MDIO interface is used transfer the management information between the core and the STA of the MAC.

Figure 1 • **CoreRMII Block Diagram**



3.1 Transmit Block

During data transmission, the transmit enable signal (MII_TXEN) is asserted active to indicate the start of an Ethernet frame, and is held active until the frame's transmission is completed. The EthernetMAC generates 4-bit MII nibble data (MII_TXD[3:0]) at 25 MHz. CoreRMII takes this nibble data and generates 2-bit data at 50 MHz on the RMII interface as output.

In 10 Mbps mode, the MAC provides 4-bit data on the MII interface on every 10^{th} clock of the 25 MHz MII clock (effectively modulating to 2.5 MHz for 10 Mbps). CoreRMII takes this nibble data and generates the 2-bit data on every 10^{th} clock of the 50 MHz RMII reference clock (effectively modulating to 5 MHz for 10 Mbps) as output.

3.2 Receive Block

During reception, the receive data valid signal (RMII_CRSDV) goes active when the frame starts, and is held active throughout the frame duration. For each clock period in which RMII_CRSDV is asserted, RMII_RXD[1:0] transfers two bits of recovered data from the PHY. In Receive mode, the PHY sends 2-bit data (RMII_RXD[1:0]) synchronous to the reference clock on the RMII interface. The RMII module detects the start-of-frame delimiter to achieve proper synchronization with the start of the frame and then forwards the 4-bit nibble wide data (MII_RXD[3:0]) to the MII interface and further to the MAC.

In 10 Mbps mode, the PHY generates the 2-bit data (RMII_RXD[1:0]) every 10^{th} clock of the reference clock on the RMII interface as output. The RMII module detects the start-of-frame delimiter to achieve proper synchronization with the start of the frame and then forwards the 4-bit nibble wide data (MII_RXD[3:0]) to the MII interface and further to the MAC.

3.2.1 Start Frame Detection (SFD)

In order to align the received data from the RMII interface with respect to the REFCLK, the core detects a pattern of alternating zeroes and ones ending with ones (D5). It uses an 8-bit shift register to capture the receive data. It then decodes the SFD pattern. Output nibbles on the MII receive interface is output on the SFD detection.

3.2.2 False Carrier

In addition, CoreRMII IP detects the false carrier condition as signaled by PHY. If the RMII PHY detects a false carrier (BAD SSD), then it generates a unique pattern of “10” rather than the normal “01” preamble pattern as output until the end of the receive end. The false carrier can occur only in the beginning of a packet where the preamble is decoded (that is, RMII_RXD[1:0] = 01). The core detects this condition and generates false carrier as output by asserting MII_RXER = ’1’, MII_RXDV = ‘0’ and MII_RXD[3:0] = ’E’ at the end of the receive cycle.

3.2.3 Error Detection

CoreRMII also provides RMII_RXERR input. The core on detection of this input asserts the MII_RXERR output in order to propagate the error.

3.2.4 Collision Detection

The core asserts MII_COL in the half-duplex mode when both the MAC and the PHY are sending valid data.

3.2.5 Carrier Sense/Data Valid

The RXDV and CRS signals are multiplexed to one signal on the RMII interface on RMII_CRSDV. The PHY asserts this signal when the receive medium is non-idle. It is asserted asynchronously on detection of carrier as per the criterion relevant to Operating mode, that is, 10/100 Mbps.

On loss of the carrier, the de-assertion is synchronous to the cycle of the REFCLK on which the first di-bit of the nibble data is presented on RMII_RXD[1:0]. If the PHY is left with few more additional bits to present on the RMII_RXD[1:0] after the de-assertion of the RMII_CRSDV signal, it then asserts the RMII_CRSDV signal on cycles of REFCLK which presents the second di-bit of the nibble data and de-asserts on the first di-bit of the nibble data. In this way, MAC can recover MII_RXDV and MII_CRS accurately.

3.2.6 Loopback

In Loopback mode, the RMII_TXD[1:0] and RMII_TXEN are connected to the RMII_RXD[1:0] and RMII_CRSDV signals respectively. The loopback feature can be used for diagnostic purposes to check the sanity of the MII and RMII interfaces. The loopback feature can be enabled by writing 1 to the loopback bit field of control register.

3.3 MDIO Interface

The MDIO interface adheres to the clause 22 of IEEE 802.3 standard. The CoreRMII registers are accessed through the MDIO interface. The MAC device driving the MDIO bus is STA. The target devices that are being managed by the MDIO interface are referred to as MDIO Manageable Devices (MMD). CoreRMII acts as a MMD.

The STA can write to the control register of the CoreRMII in order to control features of the core. The core changes the link speed, the transfer type and the loopback mode depending on the data written by STA to the control register of the core.

An Ethernet 802.3 standard clause 22 MDIO Frame Format is as shown in Figure 2.

Figure 2 • **Ethernet 802.3 Standard Clause 22 MDIO Frame Format**

	PREAMBLE	START	OPERATION	PHY ADDRESS	REG ADDRESS	TA	DATA	IDLE
READ	1111 1111 1111 1111 1111 1111 1111 1111	01	10	PPPPP	RRRRR	Z0	DDDD DDDD DDDD DDDD	Z
WRITE	1111 1111 1111 1111 1111 1111 1111 1111	01	01	PPPPP	RRRRR	10	DDDD DDDD DDDD DDDD	Z

4 Interface

4.1 I/O Signals

The port signals for the CoreRMII are listed in Table 2.

Table 2 • **CoreRMII I/O Signal Descriptions**

Port Name	Type	Description
Clocks and Resets		
TXCLKM	Input	MII transmit clock. 25 MHz
RXCLKM	Input	MII receive clock. 25 MHz
REFCLK	Input	RMII reference clock. 50 MHz
MDC	Input	Management data clock. 2.5 MHz
RSTN	Input	System reset. Active low asynchronous reset.
MII Signals		
MII_TXD[3:0]	Input	MII transmit data. From MAC to Core.
MII_TXEN	Input	MII transmit enable. From MAC to Core.
MII_RXD[3:0]	Output	MII receive data. From Core to MAC.
MII_RXDV	Output	MII receive data valid. From Core to MAC.
MII_RXERR	Output	MII receive error. From Core to MAC.
MII_COL	Output	Collision, considered asynchronous. From Core to MAC.
MII_CRS	Output	Carrier sense, considered asynchronous. From Core to MAC.
RMII Signals		
RMII_TXD[1:0]	Output	TXD [1:0] contains two bit RMII data from Core to PHY when transmitenable is HIGH. The data rate is double than that of MAC data rate for nibble data transmit.
RMII_TXEN	Output	RMII transmit enable output from Core to PHY.
RMII_RXD[1:0]	Input	RXD [1:0] contains two bit RMII data from PHY to Core when receivedata valid is HIGH. The data rate is double compared to the MAC data rate for nibble data receive.
RMII_CRSDV	Input	RMII carrier sense or data valid (shared signal) from PHY to Core.
RMII_RXERR	Input	RMII receive error from PHY to Core.
MDIO Signals		
MDO	Input	Management data output. From STA to Core.
MDO_EN	Input	Management data output enable. From STA to Core.
MDI_EXT	Input	Management data input from external PCS/PHY.
MDI	Output	Management data input. From Core to STA.

4.2 Core Parameters

4.2.1 CoreRMII Configurable Options

The configurable parameters of CoreRMII is listed in Table 3. If a configuration other than the default is required, use the configuration dialog box in SmartDesign to select appropriate values for the configurable options.

Table 3 • **CoreRMII Configuration Options**

Parameter Name	Valid Range	Default	Description
MDIO_PHYID	0 to 31	0	MDIO PHY address. Address of the core on the management interface. **Note:** MSS Hard MAC user to avoid the address 30.

5 Timing Diagrams

Figure 3 represents the packet transmitted at 100 Mbps. CoreRMII takes the 4-bit nibble data at 25 MHz on the MII interface and generates the 2-bit data at 50 MHz on the RMII interface as output.

Figure 3 • **Packet Transmitted**

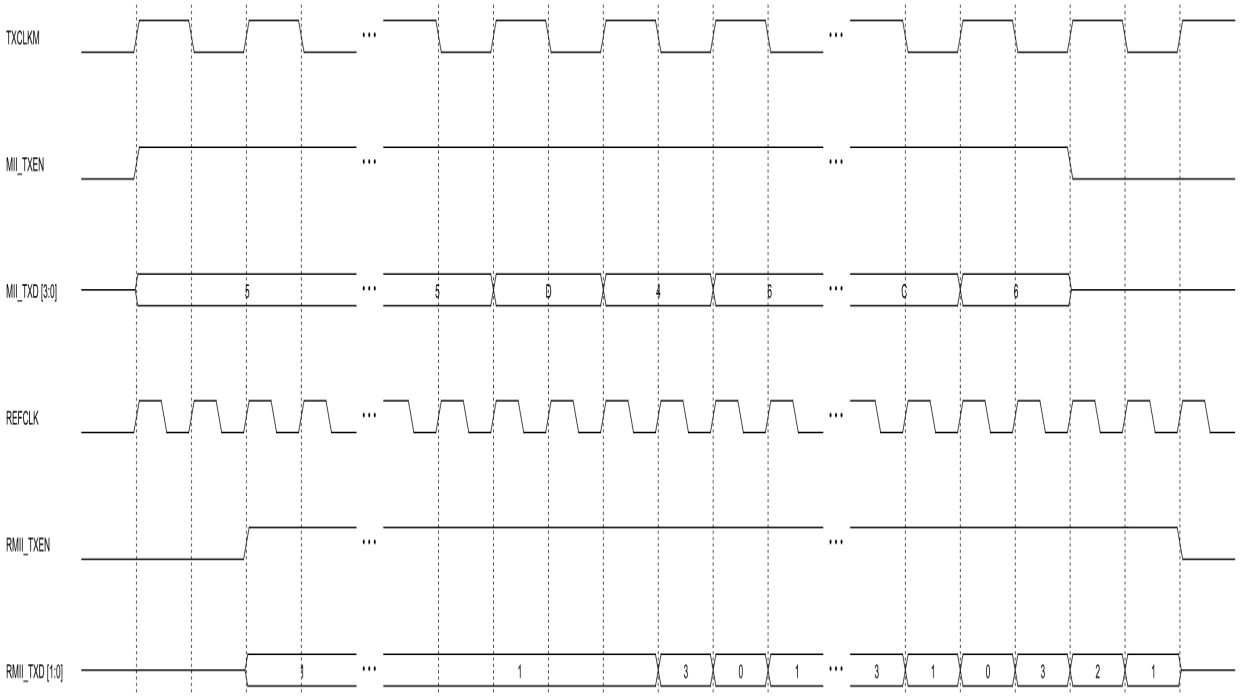


Figure 4 represents the packet received at 100 Mbps. CoreRMII takes the 2-bit nibble data at 50 MHz on the RMII interface and generates the 4-bit nibble data at 25 MHz on the MII interface as output.

Figure 4 • **Packet Received**

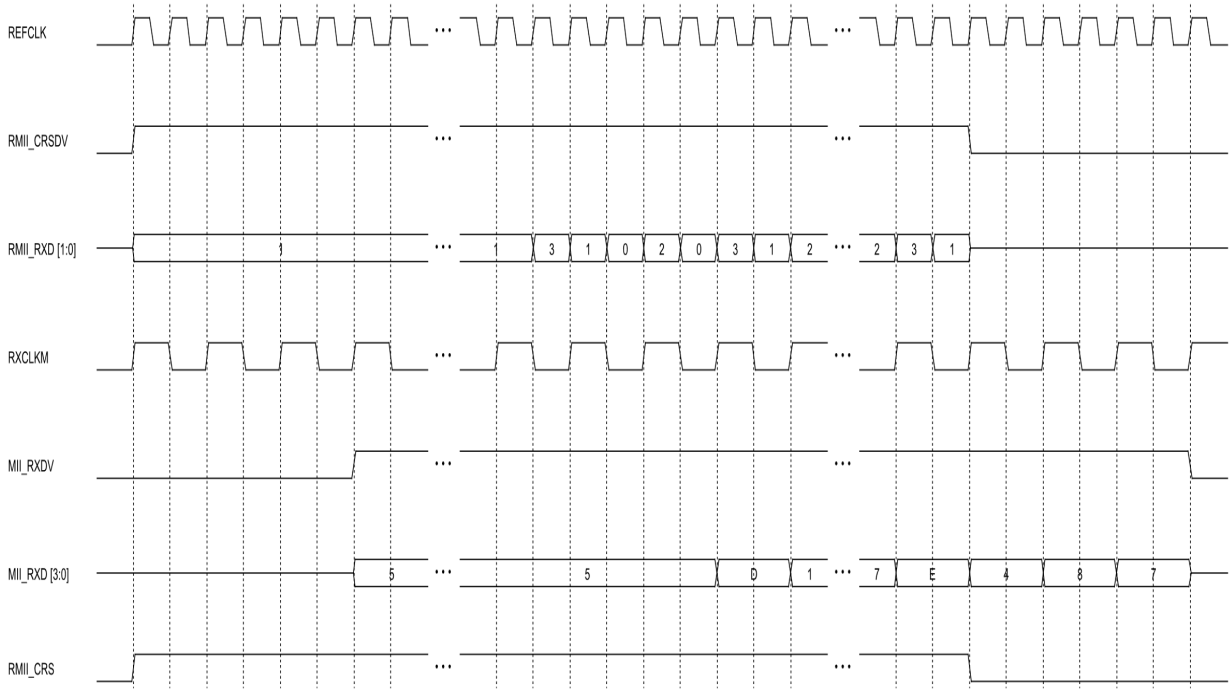
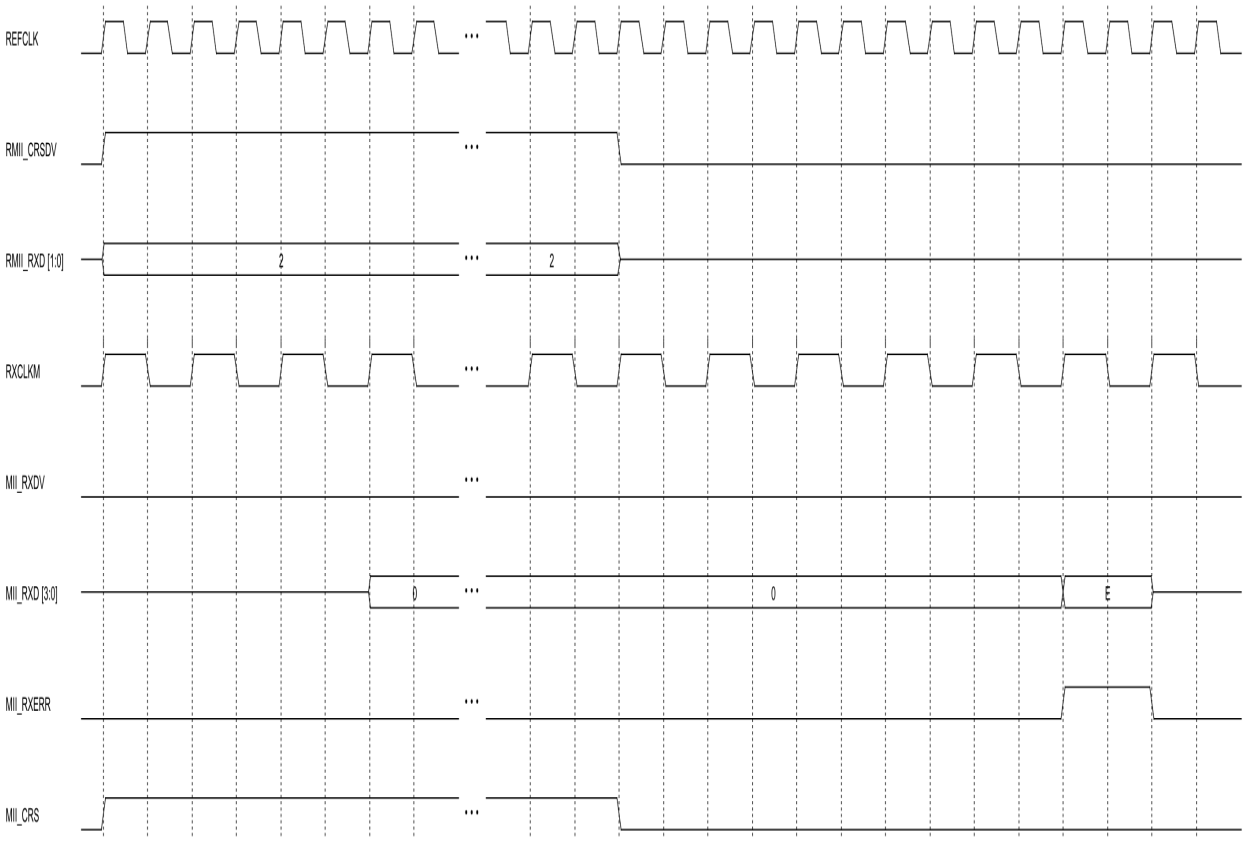


Figure 5 represents the false carrier condition on the RMII RXD signal. CoreRMII generates the false carrier code of MII_RXDV = 0, MII_RXD[3:0] = E, and MII_RXERR = 1 as output at the end of the false carrier activity.

Figure 5 • **False Carrier Condition**



6 Tool Flow

6.1 License

No license is required to use this core.

6.1.1 RTL

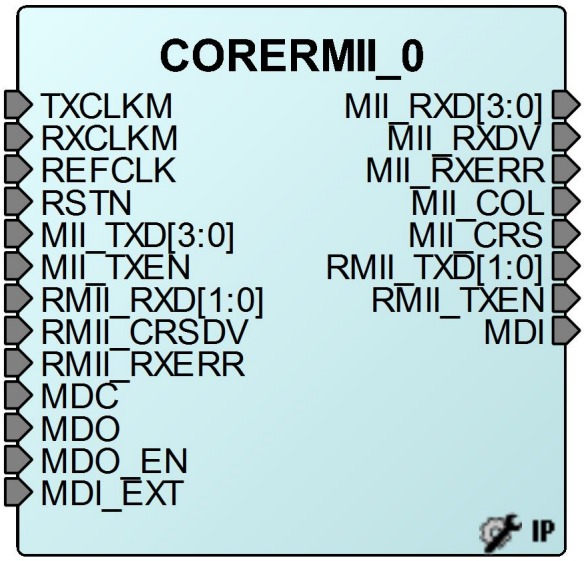
Complete RTL source code is provided for the core and testbench.

6.2 SmartDesign

CoreRMII is preinstalled in the SmartDesign IP Deployment design environment or downloaded from the online repository. Figure 6 shows an example instantiated.

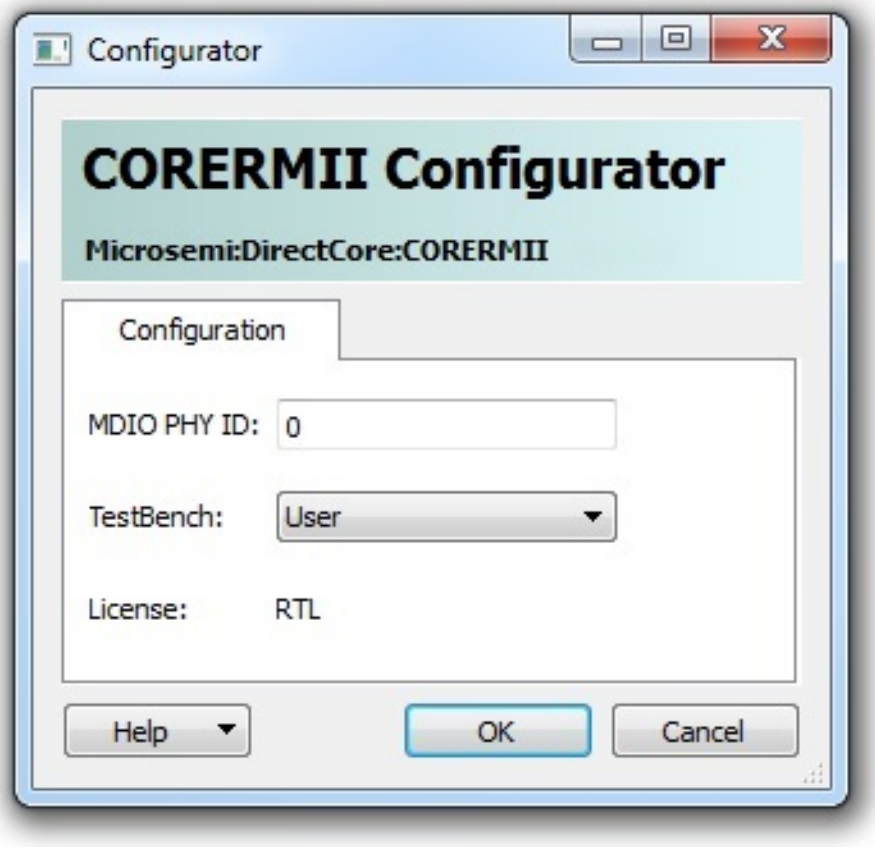
To know how to create SmartDesign project using the IP cores, refer to *Libero SoC documents page* and use the latest SmartDesign user guide.

Figure 6 • **SmartDesign CoreRMII Instance View**



The core is configured using the configuration GUI within SmartDesign, as shown Figure 7.

Figure 7 • **SmartDesign CoreRMII Configuration Window**



6.3 Simulation Flows

The user testbench for CoreRMII is included in all releases.

To run simulations:

1. Select the user testbench flow within SmartDesign.
2. Click **Save and Generate** in the Generate pane. The user testbench is selected through the Core Testbench Configuration GUI.

When SmartDesign generates the Libero project, it installs the user testbench files.

To run the user testbench:

1. Set the design root to the CoreRMII instantiation in the Libero design hierarchy pane.
2. Click **Simulation** in the Libero Design Flow window. This invokes ModelSim and automatically run the simulation.

6.4 Synthesis in Libero

To run synthesis on the CoreRMII, set the design root to the IP component instance and run the synthesis tool from the Libero design flow pane.

6.5 Place-and-Route in Libero

After the design is synthesized, run the compilation and then place-and-route tool from Libero design flow pane. CoreRMII requires no special place-and-route settings.

7 Testbench

7.1 User Testbench

An example User Testbench is included with CoreRMII.

Figure 8 • **User Testbench**

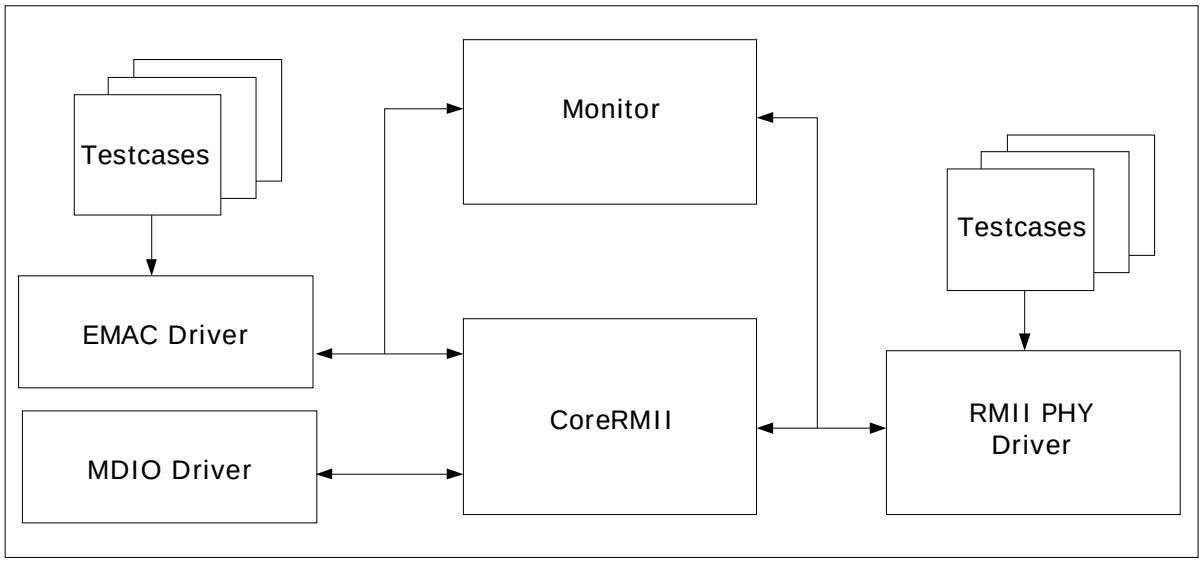


Figure 8 shows the user test bench instantiating a Microsemi® DirectCore CoreRMII design under test (DUT). The MDIO driver task drives the management information to the MDIO interface of the DUT. The DUT configures itself depending on the driven management data.

The EMAC driver tasks drives transmit transactions to the MII interface of the DUT. The DUT in turn converts the transmit transaction into corresponding transmit signals on the RMII interface. The monitor tasks check and determine whether or not the transaction is successful, and display the result.

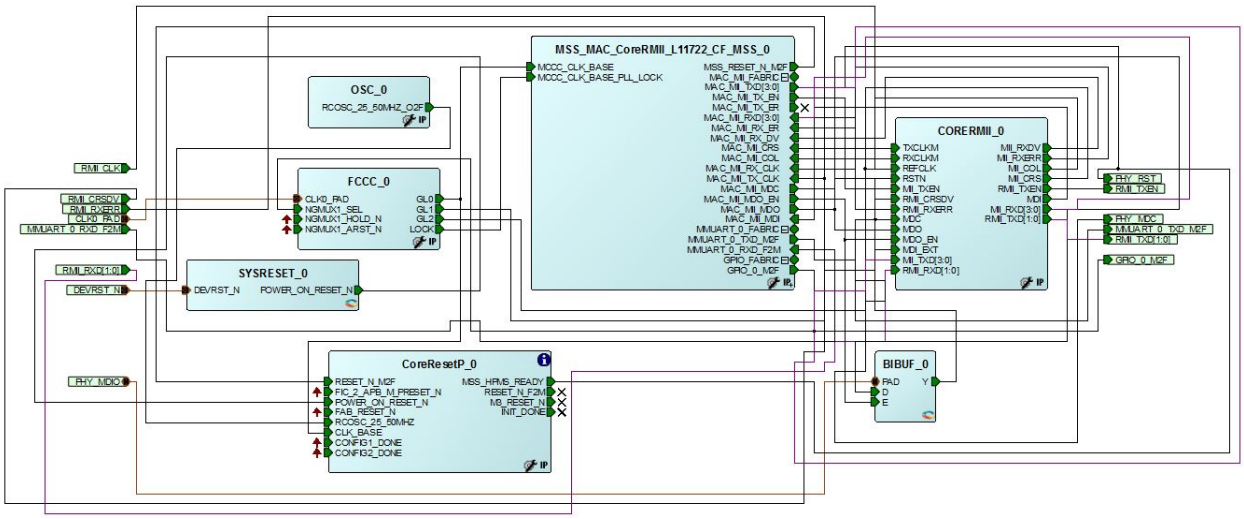
Similarly, the PHY driver tasks drive the data on the receive signals of the RMII interface of the DUT. The DUT in turn converts the transaction into corresponding receive signals on the MII interface. The monitor tasks check and determine whether or not the transaction is successful, and display the result.

8 System Integration

This section provides hints to ease the integration of CoreRMII.

- The example design described in this section contains CoreRMII, this is interfaced with the MSS_MAC which is configured to operate in MII mode.
- FABRIC RESET (SYSRESET_0) is used for all resets.
- The CORERMII_0 has TXCLKM, RXCLKM and REFCLK clocks.
- TXCLKM and RXCLKM is 25 MHz clock for 10/100 Mbps speeds, and is driven from FCCC_0/GL2.
- REFCLK is 50 MHz clock and is driven from Ethernet PHY.

Figure 9 • **System Integration**



9 Register Map and Descriptions

9.1 Register Summary

The Table 4 lists the registers used in CoreRMII.

Table 4 • **CoreRMII Registers**

Address Offset	Register Name	Type	Width	Reset Value	Description
0x00	CONTROLREG	R/W	16 bits	0x03	Control register. This register can be accessed through MDIO interface. The transfer speed, transfer type and loopback mode can be controlled by writing to the appropriate bits of the register. Refer Table 5 for bit field description.

9.2 Control Register

The bit field description of CoreRMII control register is described in Table 5.

Table 5 • **CoreRMII Control Register Description**

Bits	Name	Type	Reset Value	Description
15	Soft reset	R/W	0	Active high soft reset. This bit is self-clearing. When set, it is cleared after two MDC clock cycles.
14:3	-	-	-	Reserved
2	Loopback	R/W	0	Enable or disable the loopback feature of the core: 0 – Disable loopback 1 – Enable loopback
1	Transfer type	R/W	1	0 – Half duplex mode 1 – Full duplex mode
0	Link speed	R/W	1	0 – 10 Mbit/s 1 – 100 Mbit/s