LogiCORE IP GMII to RGMII v2.0

Product Guide for Vivado Design Suite

PG160 March 20, 2013

Table of Contents

Appendix A: Migrating

Appendix B: Debugging

Appendix C: Additional Resources

Introduction

The Xilinx LogiCORE™ IP Gigabit Media Independent Interface (GMII) to Reduced Gigabit Media Independent Interface (RGMII) design provides the RGMII between RGMII-compliant Ethernet physical media devices (PHY) and the Gigabit Ethernet controller (GEM) in the Zynq®-7000 devices. This core can be used in all three modes of operation (10/100/1000 Mb/s). The Management Data Input/Output (MDIO) interface is used to determine the speed of operation. This core can switch dynamically between the three different speed modes.

Features

- Tri-speed (10/100/1000 Mb/s) operation
- Full-duplex operation
- MDIO interface to set operating speed and duplex mode by MAC

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family[1]	Zynq-7000
Supported User Interfaces	GMII
Resources	See Table 2-1
Provided with Core	
Design Files	Encrypted RTL
Example Design	GMII to RGMII with internally generated GMII clock GMII to RGMII with externally generated GMII clock
Test Bench	Demonstration Test Bench
Constraints File	Vivado: XDC
Simulation Model	Not Provided
Supported S/W Driver	N/A
Tested Design Flows[2]	
Design Entry	Vivado® Design Suite
Simulation	Mentor Graphics Questa®SIM, Vivado Simulator
Synthesis	Xilinx Synthesis Technology (XST)
Support	
Provided by Xilinx @ www.xilinx.com/support	

Notes:

1. For a complete list of supported devices, see the Vivado IP catalog.
2. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Chapter 1

Overview

The GMII to RGMII design provides the RGMII between Ethernet physical media devices and the Gigabit Ethernet controller in Zynq®-7000 devices. This core can switch dynamically between the three different speed modes of operation (10/100/1000 Mb/s).

Feature Summary

- Tri-speed operation (10/100/1000 Mb/s)

  The line speed can be changed dynamically (for example, during run time) by programming the speed bits in the control register.

- Full-duplex operation
- MDIO interface to set operating speed by MAC

  Speed settings are contained in the control register implemented within the core. MDIO transactions are used to program this control register.

Applications

The GMII to RGMII IP is designed for use with the Gigabit Ethernet hard blocks in the Zynq-7000 Processor Subsystem (PS). The two Gigabit Ethernet MAC (ENET0 and ENET1) hard blocks present in the Zynq-7000 Processor Subsystem (PS) provide a RGMII interface through the Multiplexed I/O pins (MIO) and a GMII interface through the EMIO interface to route through the Zynq-7000 Programmable Logic (PL). The GMII to RGMII IP can be used to provide an RGMII interface using the PL. For more information on the Zynq-7000 Gigabit Ethernet Controller, refer to the *Zynq-7000 All Programmable SoC Technical Reference Manual* [Ref 7].

Unsupported Features

There are no unsupported features for this core.

Licensing and Ordering Information

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado® Design Suite under the terms of the Xilinx End User License.

Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information on pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

Figure 2-1 illustrates the connection of the Gigabit Ethernet Controller in the Zynq®-7000 PS to the GMII to RGMII IP.

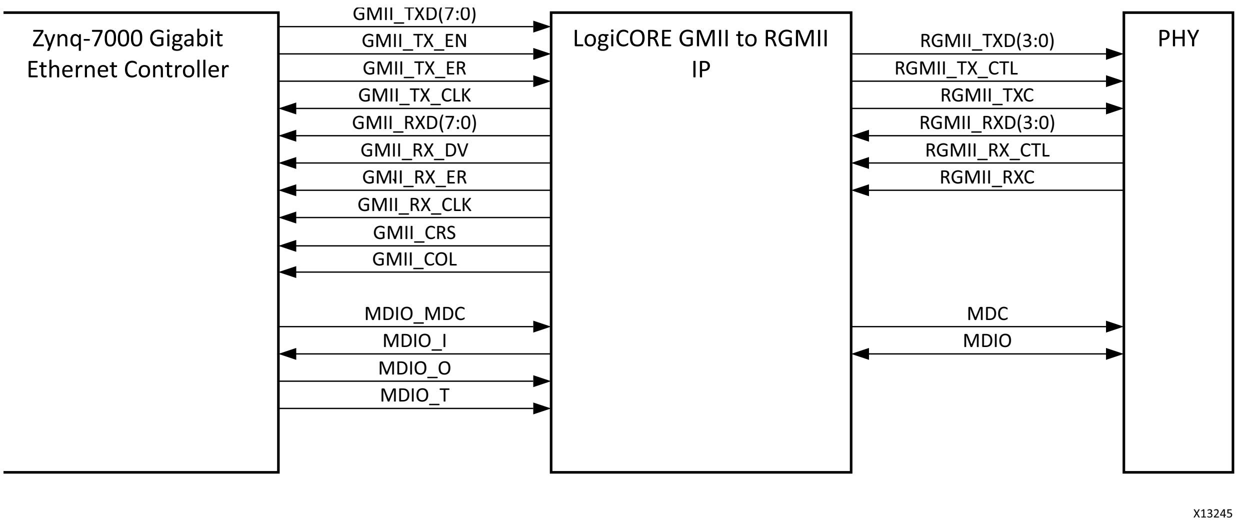


Figure 2-1: **GMII to RGMII Core Ports and Interfaces**

IMPORTANT: The MDIO interface is necessary for the operation of the core because the auto-negotiated speed of operation from the PHY is communicated to the MAC through MDIO.

The 200 MHz clock input is used as a reference clock for the IDELAY control elements and input for the management modules.

If the GMII clock is sourced internally (C_EXTERNAL_CLOCK = 0), then it is the input clock to the MMCM from which the TX clocks for all line rates (125/12.5/2.5 MHz for 1000/100/10 Mb/s, respectively).

Standards

- Ethernet standard 802.3-2008 Clauses 22 and 35
- Reduced Gigabit Media Independent Interface (RGMII) V2.0

Performance

This section describes the performance of the GMII to RGMII core.

Maximum Frequencies

The GMII to RGMII core operates at 125 MHz.

The Management Module operates at 200 MHz.

Latency

The following measurements are for the core only and do not include any IOB registers.

Transmit Path

As measured from a data octet input into `gmii_txd[7:0]` of the transmitter side of GMII interface until that data appears on the `rgmii_txd[3:0]` on the RGMII interface, the latency through the core through transmit direction is 1 clock period of `gmii_tx_clk_int`.

Receive Path

Measured from a data octet input into `rgmii_rxd[3:0]` of the receiver side of RGMII interface until that data appears on the `gmii_rxd[7:0]` on the GMII interface, the latency through the core through receive direction is 1 clock period of `rgmii_rx_clk`, plus the additional delay equal to the fixed delay specified on IDELAY component.

Throughput

The GMII to RGMII core operates at full line rates of 10/100/1000 Mb/s

Power

No information is currently provided for this core.

Resource Utilization

Resources required for the GMII to RGMII core have been estimated for the Zynq-7000 FPGAs (Table 2-1). These values were generated using Vivado® IP Catalog. They are derived from post-implementation reports. .

Table 2-1: **Device Utilization**

Parameter Values	Device Resources					
GMII Clock Source	**Slices**	**LUTs**	**FFs**	**BUFGCTRLs**	**BUFs**	**MMCMs**
Internal	134	56	77	4	2	1
External	134	56	77	3	1	0

Port Descriptions

This section describes the ports for the GMII to RGMII.

Input/Output Signals

The I/O signals for the GMII to RGMII core are listed in Table 2-2. The interfaces referenced in this table are shown in Figure 2-1.

Table 2-2: **I/O Signals**

Signal Name	Direction	Description
tx_reset	Input	Reset in TX domain
rx_reset	Input	Reset in RX domain
clkin	Input	200MHz clock is used as reference clock for the IDELAYCTRL elements and to clock the Management modules. When the GMII clock is generated internally this clock is input to the MMCM for generating 125MHz, 25MHz and 2.5MHz clocks.

Table 2-2: **I/O Signals *(Cont'd)***

Signal Name	Direction	Description
gmii_clk	Input	GMII clock from the external world. Valid when C_EXTERNAL_CLOCK = 1
gmii_tx_clk	Output	Transmit clock output from GMII to RGMII IP to Zynq PS
gmii_txd	Input	Transmit data from MAC
gmii_tx_en	Input	Data Enable control signal from MAC
gmii_tx_er	Input	Error control signal from MAC
gmii_crs	Output	Carrier sense signal (GEM does not use the Carrier Sense signal in Full Duplex mode)
gmii_col	Output	Collision signal (GEM does not use the Collision signal in Full Duplex mode)
gmii_rx_clk	Output	Receive clock output from GMII to RGMII IP to Zynq PS
gmii_rxd	Output	Received data from PHY
gmii_rx_dv	Output	Data Valid control signal from PHY
gmii_rx_er	Output	Error control signal from PHY
rgmii_txd	Output	Transmit data to PHY
rgmii_tx_ctl	Output	Control signal to PHY
rgmii_txc	Output	Clock to PHY
rgmii_rxd	Input	Received data from PHY
rgmii_rx_ctl	Input	Control signal from PHY
rgmii_rxc	Input	Clock from PHY
link_status	Output	Link status from the in-band control signal
clock_speed	Output	Clock speed from the in-band control signal
duplex_status	Output	Duplex status from the in-band control signal
mdio_i	Output	The MDIO_I line driven by Zynq PS GEM. It is used by the core during Register write operation.
mdio_o	Input	The MDIO_O line to the Zynq PS GEM.
mdio_t	Input	The MDIO_T line driven by the Zynq PS GEM.
mdio_mdc	Input	MDIO clock
mdc	Output	MDIO clock to the external PHY device
mdio	Input/Output	MDIO line which connects to the external PHY device
speed_mode[1:0]	Output	speed mode of the MAC to the clock generator 00: 10Mb/s 01: 100Mb/s 10: 1 Gb/s 11: Reserved

Interfaces

Figure 2-2 shows the ports and interfaces for the GMII to RGMII core.

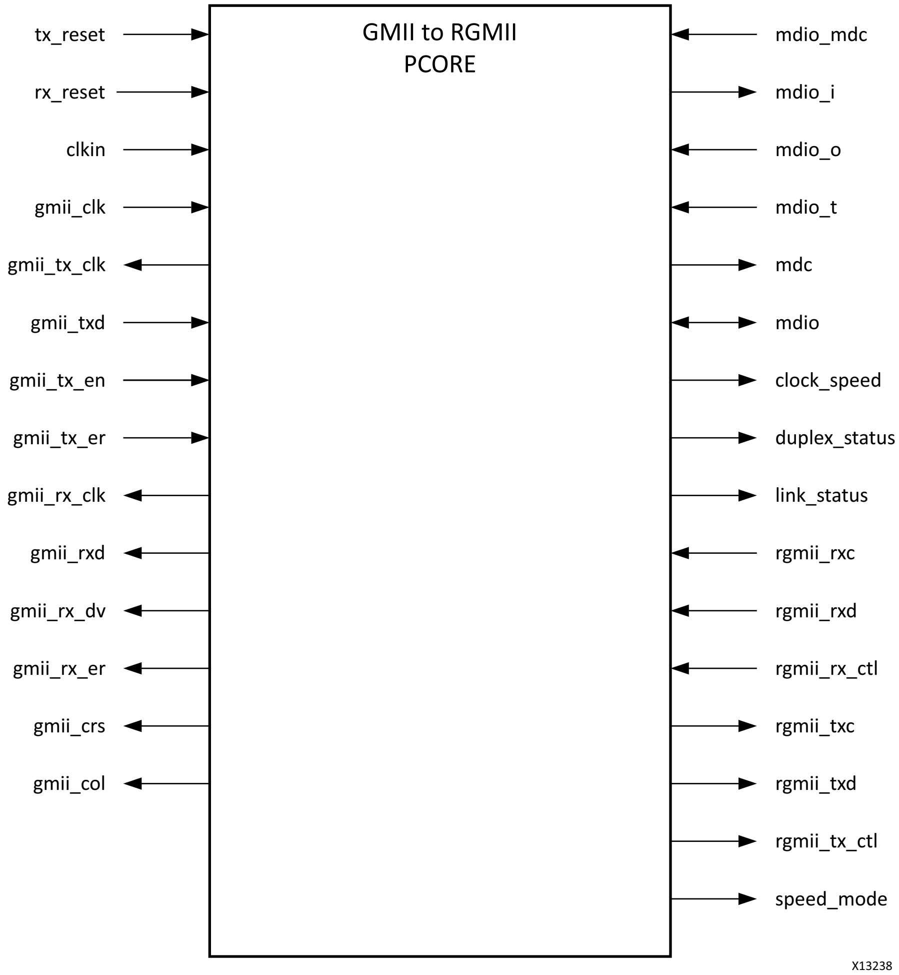


Figure 2-2: **Ports and Interfaces**

Register Space

A control register is implemented in the core which allows the software to communicate the line-rate information to the core. This allows the core to dynamically adapt to the line-rate changes. Access to this register is through the MDIO interface. The software has to initiate a MDIO read/write cycle to read from and write to this register. The software must use a PHY address which is different from the PHY address used to address the onboard PHY. The PHY address for the core can be set though the VHDL generic C_PHYADDR.

Table 2-3 illustrates the implementation of the register space in the core. This register is within the management module. The management module monitors the MDIO_O line for a new MDIO cycle. When a new cycle is initiated and the PHY address matches the PHY address assigned to this core, the module latches the data-field of the MDIO frame to the control register for a write cycle or muxes out the content of the control register to the MDIO_I line for a read cycle.

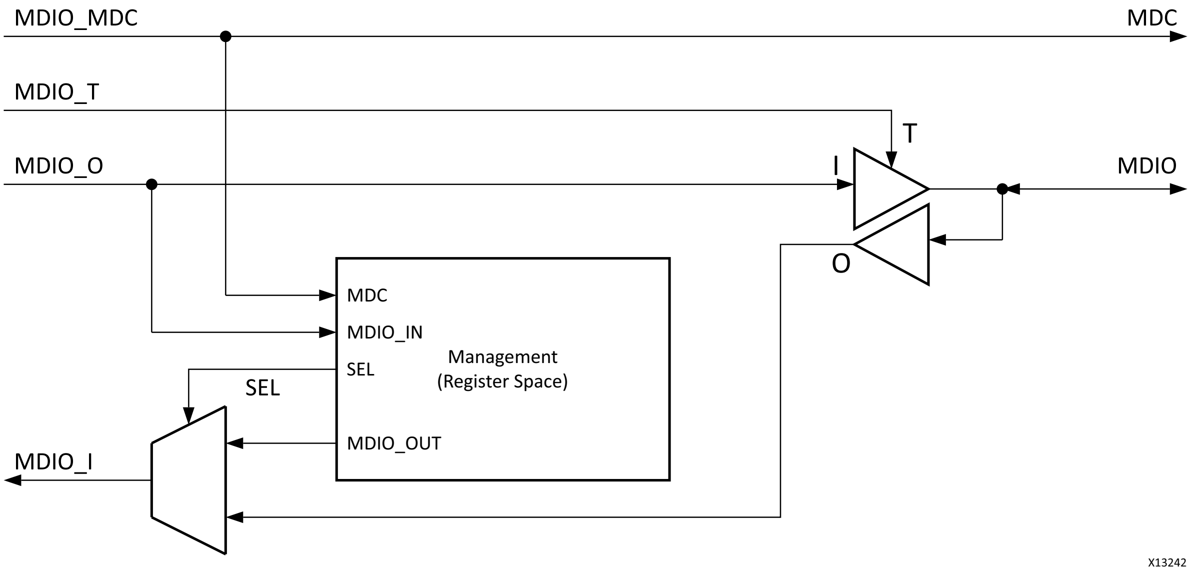


Figure 2-3: **Register Space Implementation**

Control Register

This register is 16-bits wide. Its address is 0x10. The composition of this register is similar to the IEEE standard 802.3 MDIO control register 0x0, which is shown in Table 2-3.

Table 2-3: **Core-Specific Control Register (Address 0x10)**

Bit(s)	Name	Description	R/W
15	Reset	1 = Resets the core and this register 0 = Normal operation	R/W Self-clearing
14	Reserved	Write as 0, ignore on read	R/W
13	Speed Selection (LSB)	6 13 1 1 = Reserved 1 0 = 1000 Mb/s 0 1 = 100 Mb/s 0 0 = 10 Mb/s	R/W
12:9	Reserved	Write as 0, ignore on read	R/W
8:7	Reserved	Write as 0, ignore on read	R/W

Table 2-3: **Core-Specific Control Register (Address 0x10)** ***(Cont'd)***

Bit(s)	Name	Description	R/W
6	Speed Selection (MSB)	6 13 1 1 = Reserved 1 0 = 1000 Mb/s 0 1 = 100 Mb/s 0 0 = 10 Mb/s	R/W
5:0	Reserved	Write as 0, ignore on read	R/W

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

General Design Guidelines

The Xilinx Synthesis Tool and Vivado® Design Suite are used to synthesize the GMII to RGMII core.

The parameters defined in Table 3-1 allow you to obtain a GMII to RGMII design that is uniquely tailored for your system.

Table 3-1: **Design Parameters**

Parameter Name	Description	Default value	Allowable values
C_EXTERNAL_CLOCK	Select GMII clock source	0	0 = GMII clock is internally generated 1 = GMII clock is sourced externally
C_PHYADDR	5-bit address used to identify the core in a MDIO transaction.	8 (01000 binary)	00000 binary to 11111 binary. The PHY address used here must be different from the address assigned to the onboard PHY.

Clocking

The GMII clock is used by the transmit circuitry of the core. This clock can be sourced externally or can be generated internally within the core. The VHDL generic C_EXTERNAL_CLOCK is used to indicate this choice.

Sourced Externally (C_EXTERNAL_CLOCK = 1)

This mode is active when the VHDL generic C_EXTERNAL_CLOCK is set to 1. Ensure that the GMII clock frequency is appropriate for the line rate. That is, it should be 2.5 MHz for 10Mb/

s, 25 MHz for 100Mb/s and 125 MHz for 1000 Mb/s. The external clock is routed to the global clock network though BUFG before being used in the design. See Figure 3-1.

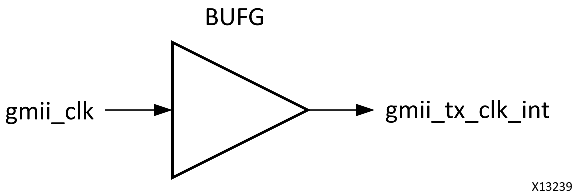


Figure 3-1: **External GMII Clock**

Generated Internally (C_EXTERNAL_CLOCK = 0)

This mode is active when the VHDL generic C_EXTERNAL_CLOCK is set to 0. The GMII to RGMII IP has a built-in clock generator for providing 2.5MHz, 25MHz, and 125MHz frequency clocks for 10Mb/s, 100Mb/s, and 1Gb/s speeds of operation, respectively. The clock generator uses an MMCM and two BUFGMUXs to switch between three different clock frequencies for the three different speed modes of the MAC. The *speed_mode(0)* and *speed_mode(1)* signals are used as selection pins of BUFGMUX0 and BUFGMUX1 respectively as shown in Figure 3-2.

The output (*gmii_tx_clk_int*) of BUFGMUX1 is used to provide the *gmii_tx_clk* clock to GEM as illustrated in Figure 3-3. The *rgmii_txc* clock is generated from *gmii_tx_clk_int* using ODDR as illustrated by Figure 3-3. The *rgmii_rxc* clock is driven through BUFR and BUFG to provide the *gmii_rx_clk* clock to GEM as illustrated in Figure 3-3.

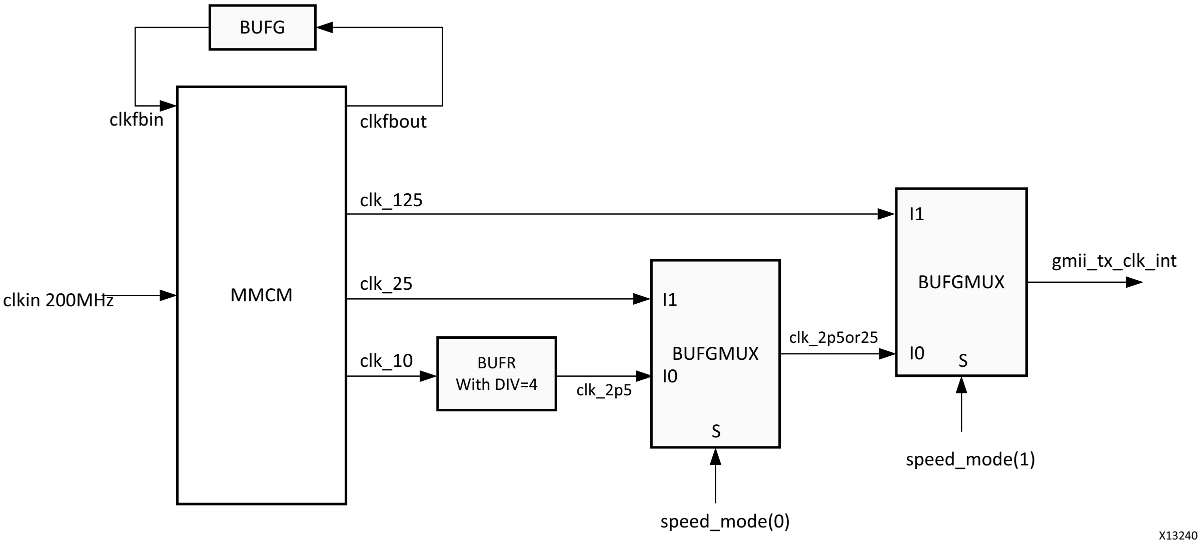


Figure 3-2: **Clock Generator**

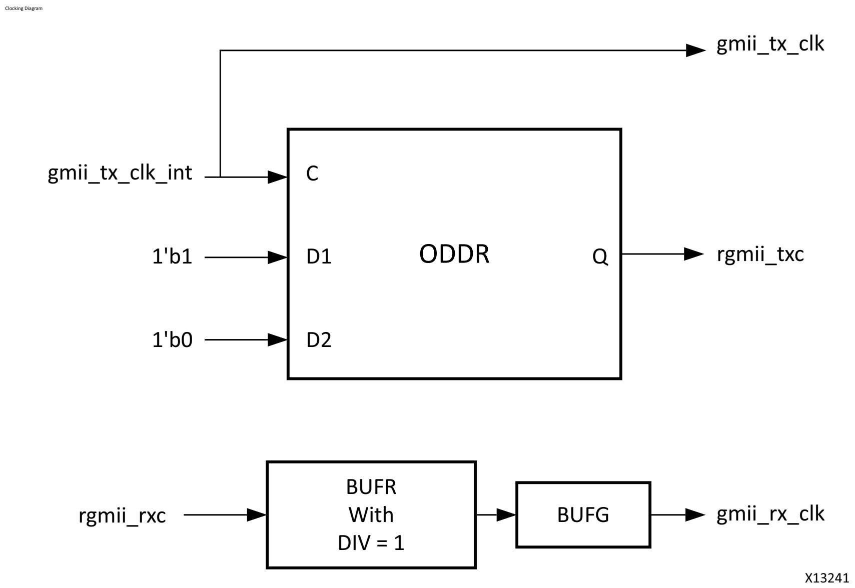


Figure 3-3: **Clocking Diagram**

Resets

Figure 3-4 shows the reset structure for the GMII to RGMII core.

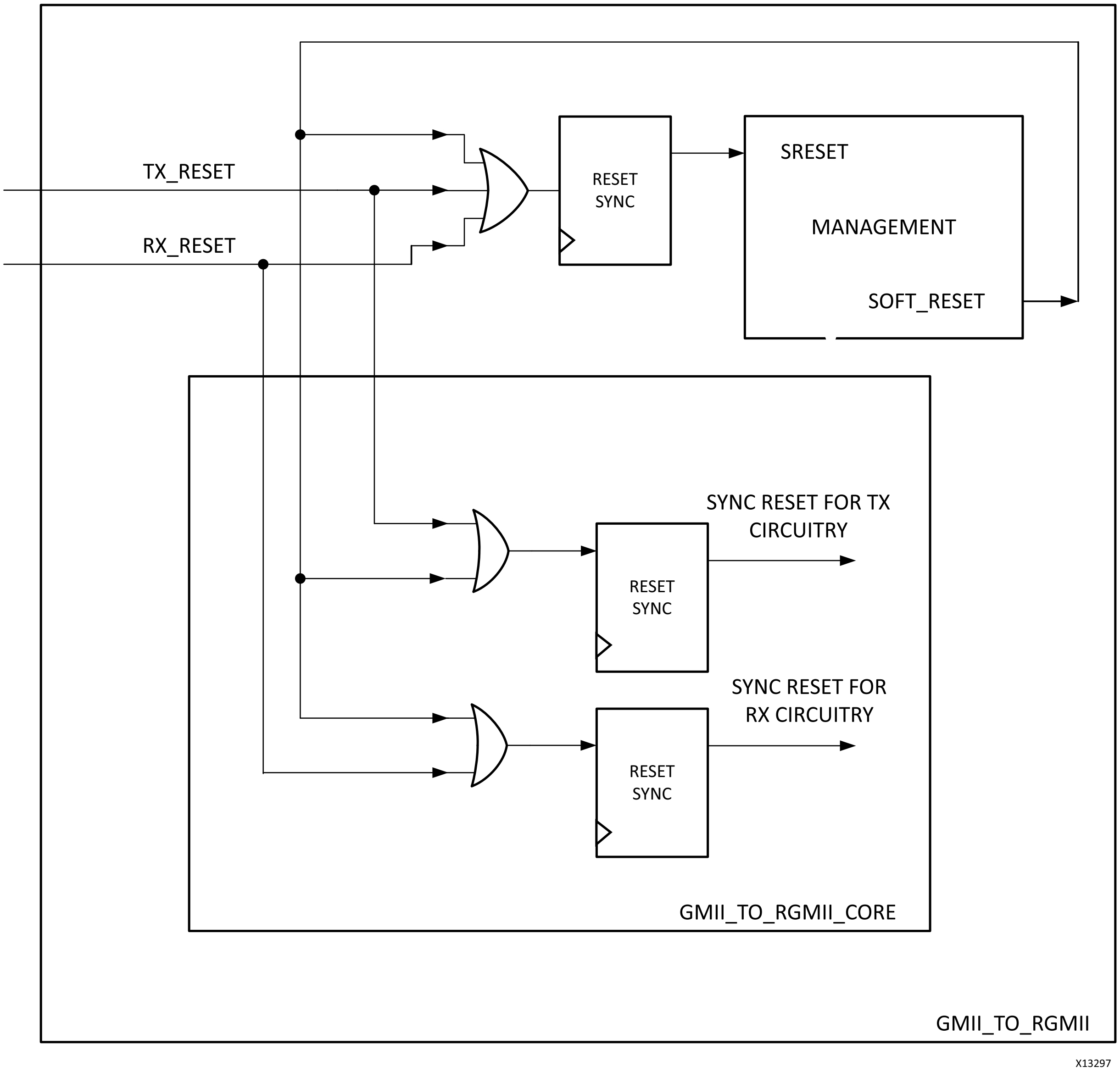


Figure 3-4: **GMII to RGMII Reset Structure**

Protocols

GMII Transmission

This section includes figures that illustrate GMII transmission.

Normal Frame Transmission

Normal outbound frame transfer timing is illustrated in Figure 3-5. This figure shows that an Ethernet frame is preceded by an 8-byte preamble field (inclusive of the Start of Frame Delimiter (SFD), and completed with a 4-byte Frame Check Sequence (FCS) field. This frame is created by the MAC connected to the other end of the GMII.

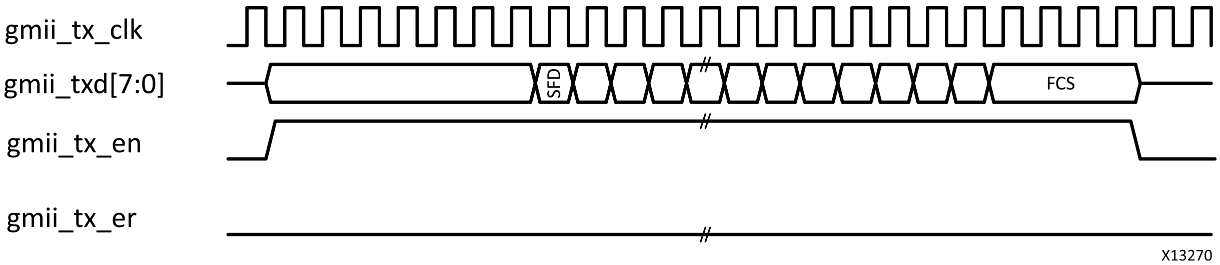


Figure 3-5: **Normal Frame Transmission**

Error Propagation

A corrupted frame transfer is illustrated in Figure 3-6. An error can be injected into the frame by asserting `gmii_tx_er` at any point during the `gmii_tx_en` assertion window.

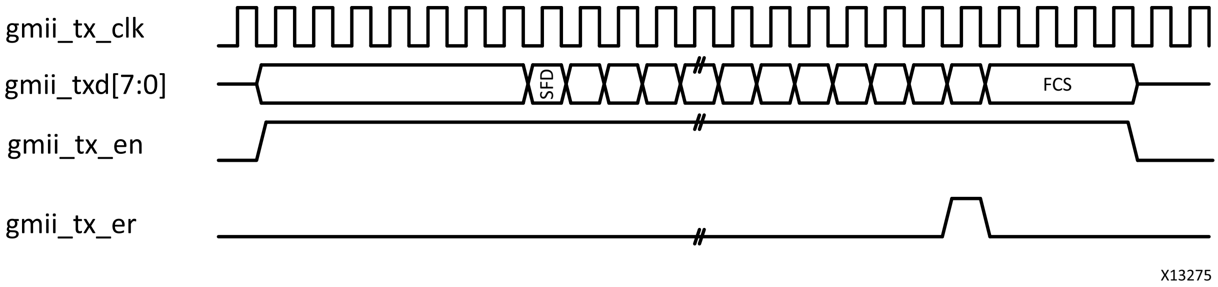


Figure 3-6: **GMII Error Propogation Within a Frame**

GMII Reception

This section includes figures that illustrate GMII reception.

Normal Frame Reception

The timing of normal inbound frame transfer is illustrated in Figure 3-8 This shows that Ethernet frame reception is preceded by a preamble field. The *IEEE 802.3-2008* Specification (see clause 35) allows for up to all of the seven preamble bytes that precede the Start of Frame Delimiter (SFD) to be lost in the network. The SFD is always present in well-formed frames.

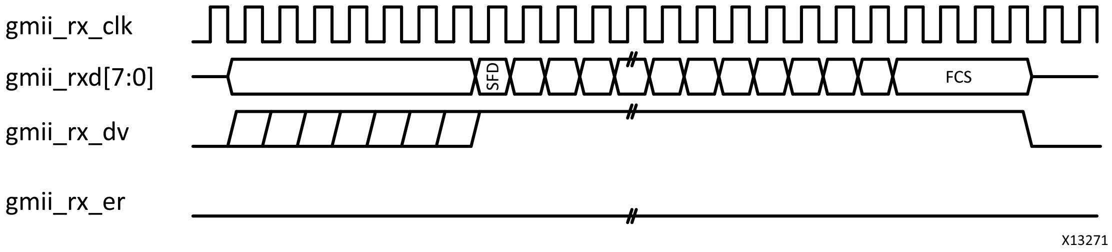


Figure 3-7: **GMII Normal Frame Reception**

Frame Reception with Errors

The signal `gmii_rx_er` when asserted within the assertion window signals that a frame was received with a detected error (Figure 3-7).

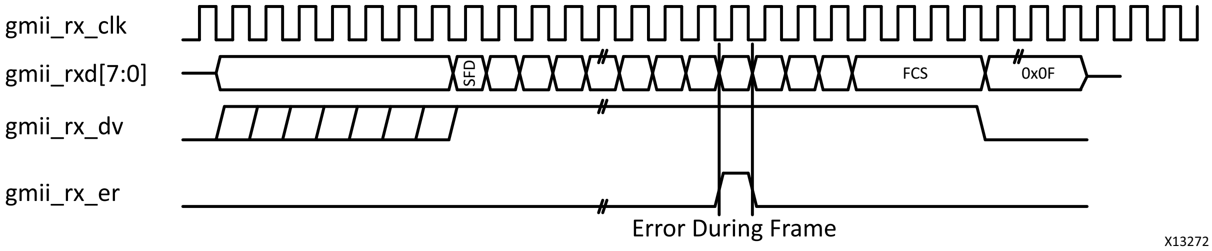


Figure 3-8: **GMII Frame Reception with Errors**

MII Transmission - 10/100 Mb/s Frame

The operation is similar to GMII transmission. In this case only the lower 4 bits (`gmii_txd[3:0]`) are valid.

MII Reception - 10/100 Mb/s Frame

The operation is similar to GMII reception. In this case only the lower 4 bits (`gmii_rxd[3:0]`) are valid.

RGMII Interface Protocols

The RGMII is intended as an alternative to the IEEE Std 802.3-2008 Clauses 22 and 35 (MII) and Clauses 34–39, 41–42 (GMII), and the TBI. The principle objective is to reduce the number of pins required to interconnect the MAC and the PHY from a maximum of 28 pins (TBI) to 12 pins in a cost effective and technology independent manner. To accomplish this objective, the datapaths and all associated control signals are reduced and control signals multiplexed together, and both edges of the clock are used. For Gigabit operation, the clocks operate at 125MHz, and for 10/100 operation, the clocks operate at 2.5MHz and 25MHz, respectively.

RGMII Transmission and Reception

Normal outbound frame transfer timing is illustrated in Figure 3-9.

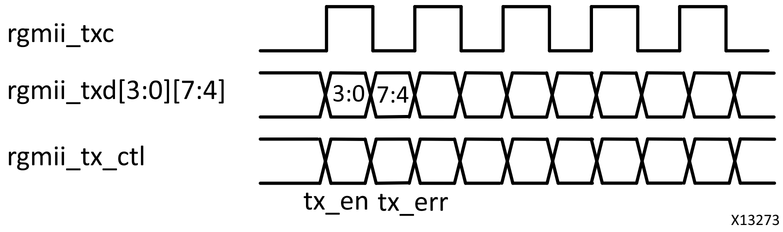


Figure 3-9: **RGMII Normal Frame Transmission**

Normal inbound frame transfer timing is illustrated in Figure 3-10.

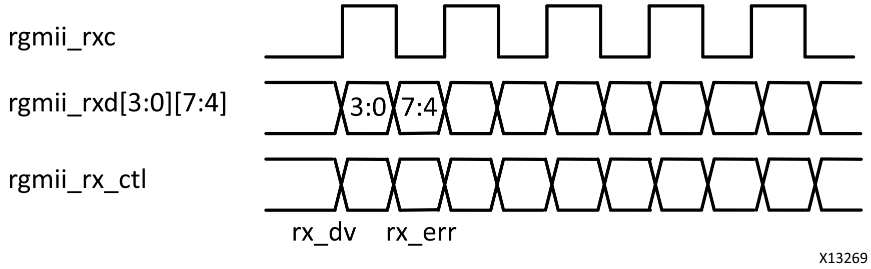


Figure 3-10: **RGMII Normal Frame Reception**

Multiplexing of data and control information is done by using both edges of the reference clocks and sending the lower 4 bits on the rising edge and the upper 4 bits on the falling edge. Control signals can be multiplexed into a single clock cycle using the same technique.

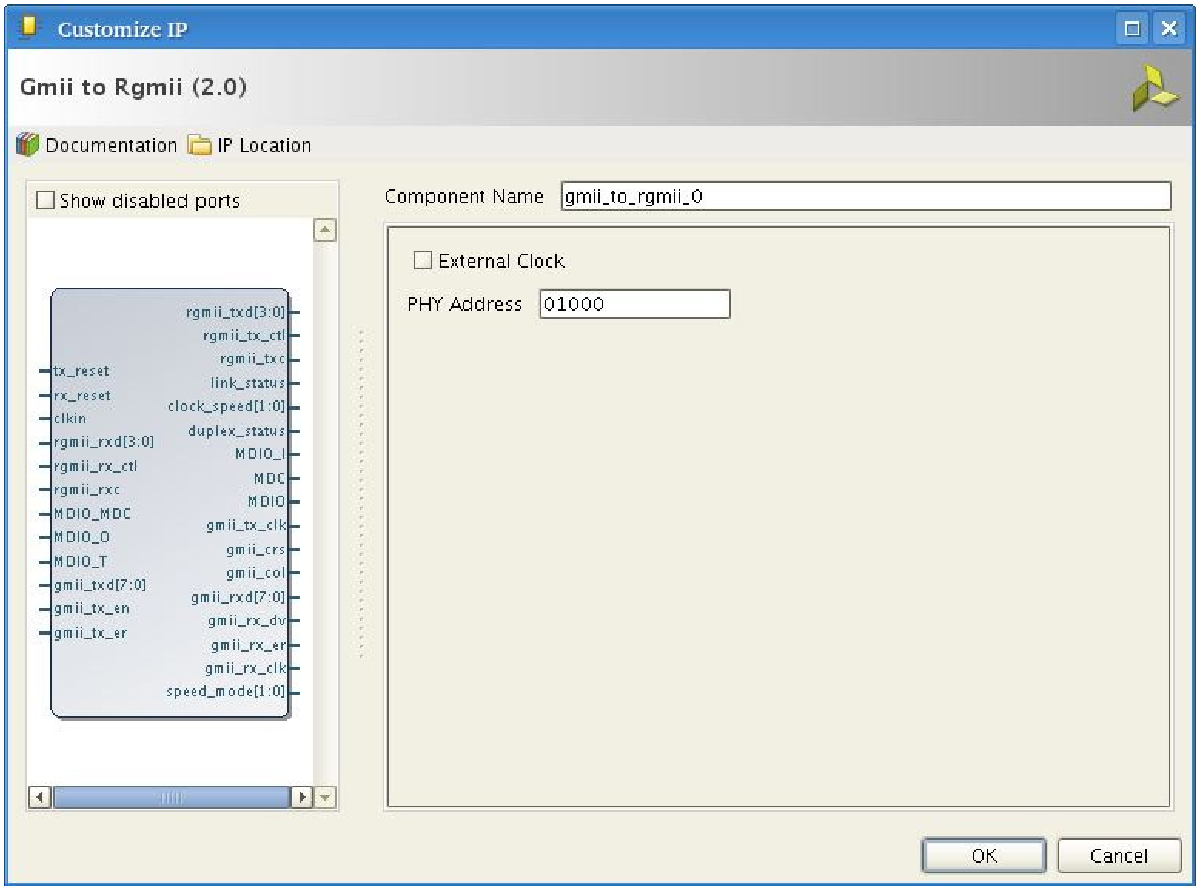
Customizing and Generating the Core

This chapter includes information about using Xilinx tools to customize and generate the core in the Vivado® Design Suite.

Vivado Integrated Design Environment (IDE)

The GMII to RGMII core is generated using the Vivado IP catalog.

Figure 4-1 displays the GMII to RGMII customization screen used to set core parameters and options.



□ Show disabled ports

Component Name: gmii_to_rgmii_0

□ External Clock

PHY Address: 01000

Figure 4-1: **Core Customization Screen**

Component Name

The component name is used as the base name of the output files generated for the core. Names must begin with a letter and can be composed of the following characters: a through z, 0 through 9, and the underscore (_).

External Clock

Select this option to source the GMII clock externally. When selected, ensure that the GMII clock frequency is appropriate for the line rate: 2.5 MHz for 10Mb/s, 25 MHz for 100Mb/s and 125 MHz for 1000 Mb/s.

By default, the GMII clock is generated internally. The GMII to RGMII IP has a built-in clock generator for providing 2.5MHz, 25MHz, and 125MHz frequency clocks for 10 Mb/s, 100 Mb/s and 1000 Mb/s speeds of operation, respectively.

PHY Address

The PHY Address is the 5-bit address used to identify the core in a MDIO transaction. Valid ranges are 00000 binary to 11111 binary. The PHY address here must be different from the address assigned to the onboard PHY.

Output Generation

The GMII to RGMII solution delivers files to several file groups. By default, the file groups required for use of the GMII to RGMII or for opening the IP Example design are generated when the core is generated.

The file groups generated are found in the IP Sources tab of the Sources window, where they are listed for each IP in the project. The file groups available for the GMII to RGMII solution are described in Table 4-1.

Table 4-1: **File Groups**

Group	Description
Examples	Includes all source required to be able to open, implement and simulate the IP example design project: the example design HDL, the example design XDC file, and the demonstration test bench HDL.
Synthesis	Includes all synthesis sources required by the core. For the GMII to RGMII solution, this includes both encrypted and unencrypted sources. Only unencrypted sources are visible.
Instantiation Template	Example instantiation template.

Chapter 5

Constraining the Core

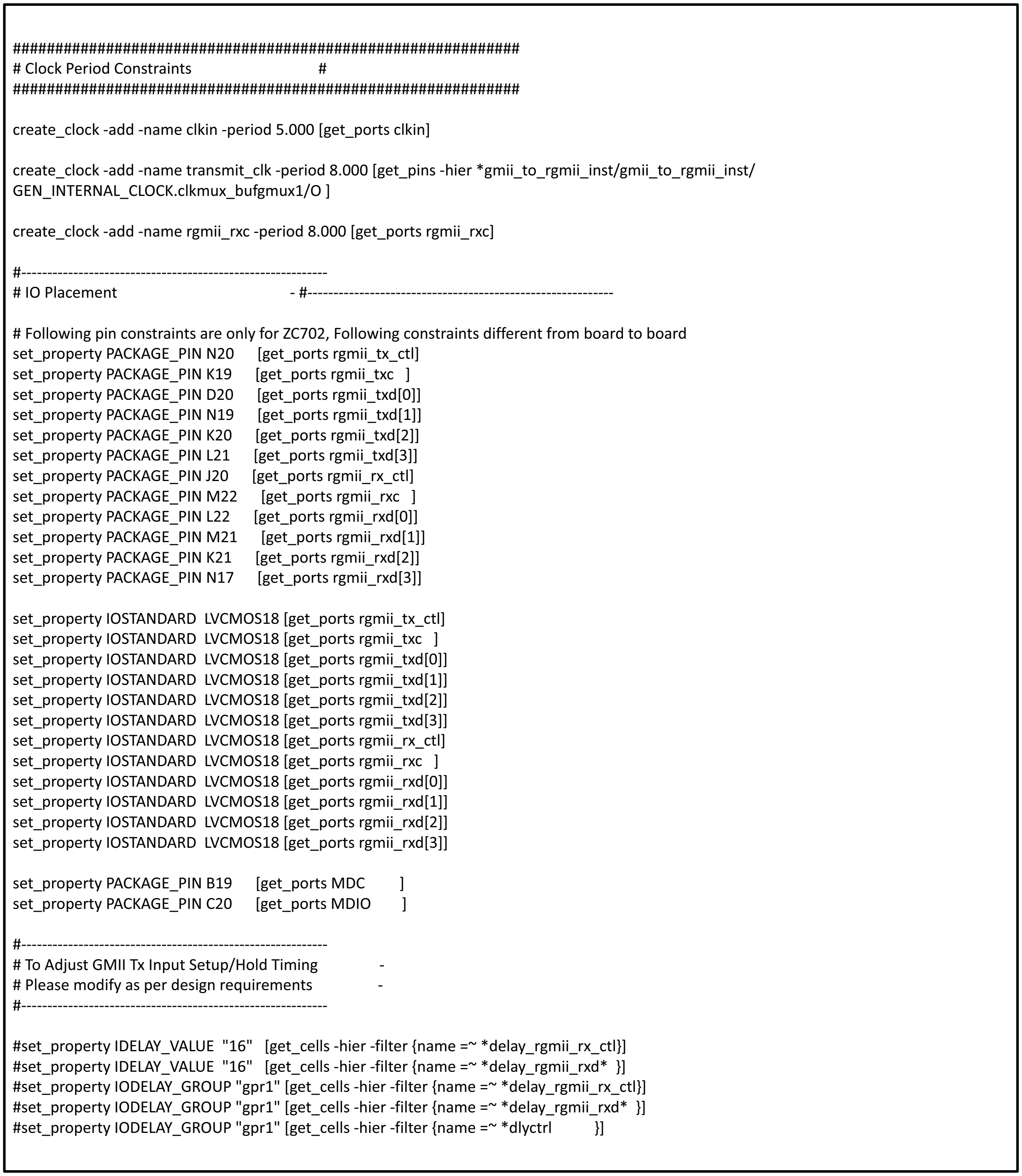
This chapter contains information about constraining the core in the Vivado® Design Suite environment.

Required Constraints

The GMII to RGMII core requires design constraints to guarantee performance.

Vivado Design Suite

These constraints should be placed in an XDC of the top level of the design. The example of the constraint text shown in Figure 5-1 is based on the port names of the GMII to RGMII 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.



```
################################################################
# Clock Period Constraints                              #
################################################################

create_clock -add -name clkin -period 5.000 [get_ports clkin]

create_clock -add -name transmit_clk -period 8.000 [get_pins -hier *gmii_to_rgmii_inst/gmii_to_rgmii_inst/
GEN_INTERNAL_CLOCK.clkmux_bufgmux1/O ]

create_clock -add -name rgmii_rxc -period 8.000 [get_ports rgmii_rxc]

#-----------------------------------------------------------
# IO Placement                                  - #-----------------------------------------------------------

# Following pin constraints are only for ZC702, Following constraints different from board to board
set_property PACKAGE_PIN N20     [get_ports rgmii_tx_ctl]
set_property PACKAGE_PIN K19     [get_ports rgmii_txc   ]
set_property PACKAGE_PIN D20     [get_ports rgmii_txd[0]]
set_property PACKAGE_PIN N19     [get_ports rgmii_txd[1]]
set_property PACKAGE_PIN K20     [get_ports rgmii_txd[2]]
set_property PACKAGE_PIN L21     [get_ports rgmii_txd[3]]
set_property PACKAGE_PIN J20     [get_ports rgmii_rx_ctl]
set_property PACKAGE_PIN M22     [get_ports rgmii_rxc   ]
set_property PACKAGE_PIN L22     [get_ports rgmii_rxd[0]]
set_property PACKAGE_PIN M21     [get_ports rgmii_rxd[1]]
set_property PACKAGE_PIN K21     [get_ports rgmii_rxd[2]]
set_property PACKAGE_PIN N17     [get_ports rgmii_rxd[3]]

set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_tx_ctl]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_txc   ]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_txd[0]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_txd[1]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_txd[2]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_txd[3]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rx_ctl]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rxc   ]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rxd[0]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rxd[1]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rxd[2]]
set_property IOSTANDARD  LVCMOS18 [get_ports rgmii_rxd[3]]

set_property PACKAGE_PIN B19     [get_ports MDC        ]
set_property PACKAGE_PIN C20     [get_ports MDIO       ]

#-----------------------------------------------------------
# To Adjust GMII Tx Input Setup/Hold Timing              -
# Please modify as per design requirements               -
#-----------------------------------------------------------

#set_property IDELAY_VALUE  "16"   [get_cells -hier -filter {name =~ *delay_rgmii_rx_ctl}]
#set_property IDELAY_VALUE  "16"   [get_cells -hier -filter {name =~ *delay_rgmii_rxd*  }]
#set_property IODELAY_GROUP "gpr1" [get_cells -hier -filter {name =~ *delay_rgmii_rx_ctl}]
#set_property IODELAY_GROUP "gpr1" [get_cells -hier -filter {name =~ *delay_rgmii_rxd*  }]
#set_property IODELAY_GROUP "gpr1" [get_cells -hier -filter {name =~ *dlyctrl         }]
```

X13243

Figure 5-1: **Timing Constraints**

Device, Package, and Speed Grade Selections

The core can be implemented in a Zynq®-7000 device with these attributes:

- Is large enough to accommodate the core
- Contains a sufficient number of IOBs
- Has a supported speed grade (-1 or faster)

Clock Frequencies

Following are clock frequency requirements:

- clkin - 200 MHz
- gmii_clk - (Present when the GMII clock is sourced externally) 2.5/25/125 MHz for 10/100/1000 Mb/s, respectively

Clock Management

No information is currently provided for this core.

Clock Placement

No information is currently provided for this core.

Banking

All ports should be given location constraints appropriate to your design, within banking limits.

Transceiver Placement

Not applicable. This core does not use transceivers.

I/O Standard and Placement

According to the RGMII v2.0 specification, the I/O pins must use 1.5v HSTL interface voltages. Like the majority of PHYs, those selected for Xilinx development boards are multi-standard and operate at either 1.8V or 2.5V.

Table 5-1 provides information on the I/O standards supported for different Zynq 7000 series devices that correspond to the Xilinx 7 Series Programmable Logic Equivalent.

Table 5-1: **I/O Standards Supported**

Xilinx 7 Series Programmable Logic Equivalent	RGMII Signals Voltage Level Supported			Miscellaneous PHY Signals (MDIO, MDC, RESET) Voltage Level Supported		
	3.3V	**2.5V**	**1.8V**	**3.3V**	**2.5V**	**1.8V**
Artix® 7	No[1]	Yes[2]	Yes[3]	Yes[2]	Yes[2]	Yes[3]
Kintex® 7	No[1]	Yes[2]	Yes[3]	Yes[2]	Yes[2]	Yes[3]

1. High Range (HR) I/O duty cycle distortion exceeds RGMII specification.
2. Requires the use of HR I/O.
3. Limited 1.8V RGMII-only PHY devices are available. If one of these devices is not used, external voltage level shifting logic is required.

Appendix A

Migrating

For information on migrating to the Vivado® Design Suite, see *Vivado Design Suite Migration Methodology Guide* [Ref 8].

Appendix B

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools. In addition, this appendix provides a step-by-step debugging process and a flow diagram to guide you through debugging the GMII to RGMII core.

The following topics are included in this appendix:

- Finding Help on Xilinx.com
- Debug Tools
- Simulation Debug
- Hardware Debug

On the RGMII interface, the RX data should provide sufficient setup time with regard to the RX clock so that it can be sampled correctly. The IDLEAY components are provided on the RX datapath. The correct delay value must be specified so that the RX data can be correctly sampled.

Finding Help on Xilinx.com

To help in the design and debug process when using the GMII to RGMII, the Xilinx Support web page (www.xilinx.com/support) contains key resources such as product documentation, release notes, answer records, information about known issues, and links for opening a Technical Support WebCase.

Documentation

This product guide is the main document associated with the GMII to RGMII. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page (www.xilinx.com/support) or by using the Xilinx Documentation Navigator.

Download the Xilinx Documentation Navigator from the Design Tools tab on the Downloads page (www.xilinx.com/download). For more information about this tool and the features available, open the online help after installation.

Known Issues

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can also be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as:

- Product name
- Tool messages
- Summary of the issue encountered

A filter search is available after results are returned to further target the results.

Master Answer Record for the GMII to RGMII core

AR 54689

Contacting Technical Support

Xilinx provides technical support at www.xilinx.com/support for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support:

1. Navigate to www.xilinx.com/support.
2. Open a WebCase by selecting the WebCase link located under Support Quick Links.

When opening a WebCase, include:

- Target FPGA including package and speed grade.
- All applicable Xilinx Design Tools and simulator software versions.
- Additional files based on the specific issue might also be required. See the relevant sections in this debug guide for guidelines about which files to include with the WebCase.

Debug Tools

Many tools are available to address GMII to RGMII design issues. It is important to know which tools are useful for debugging various situations.

Example Design

The GMII to RGMII core is delivered with an example design that can be synthesized, complete with functional test benches.

External Clocking

Figure B-1 illustrates the example design for top-level HDL when the GMII clock is sourced externally.

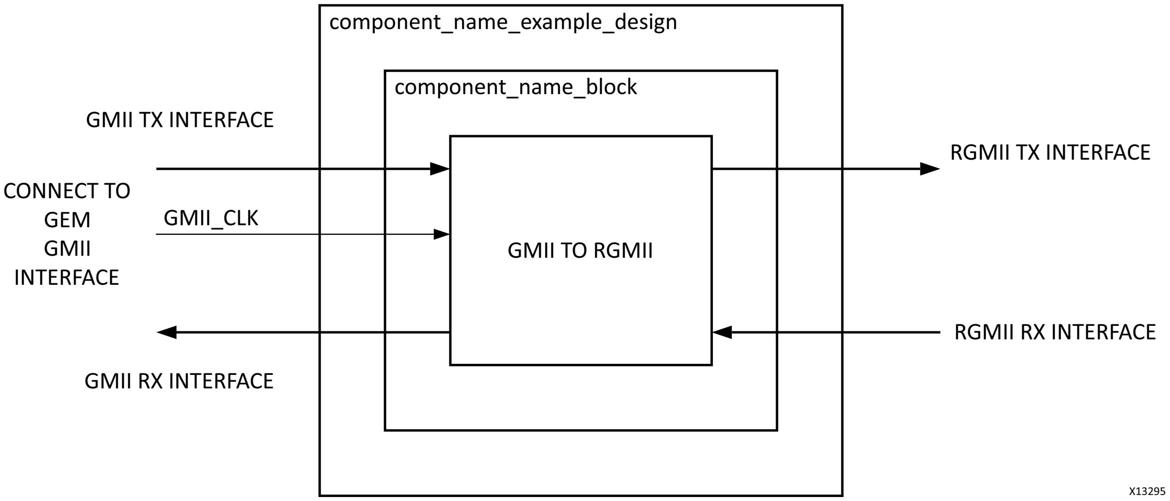


Figure B-1: **Example Design HDL with External GMII Clock**

In this case, the GMII_CLOCK is sourced externally. No additional clocking resources are used.

Internal Clocking

Figure B-2 illustrates the example design for top level HDL when the GMII clock is sourced internally.

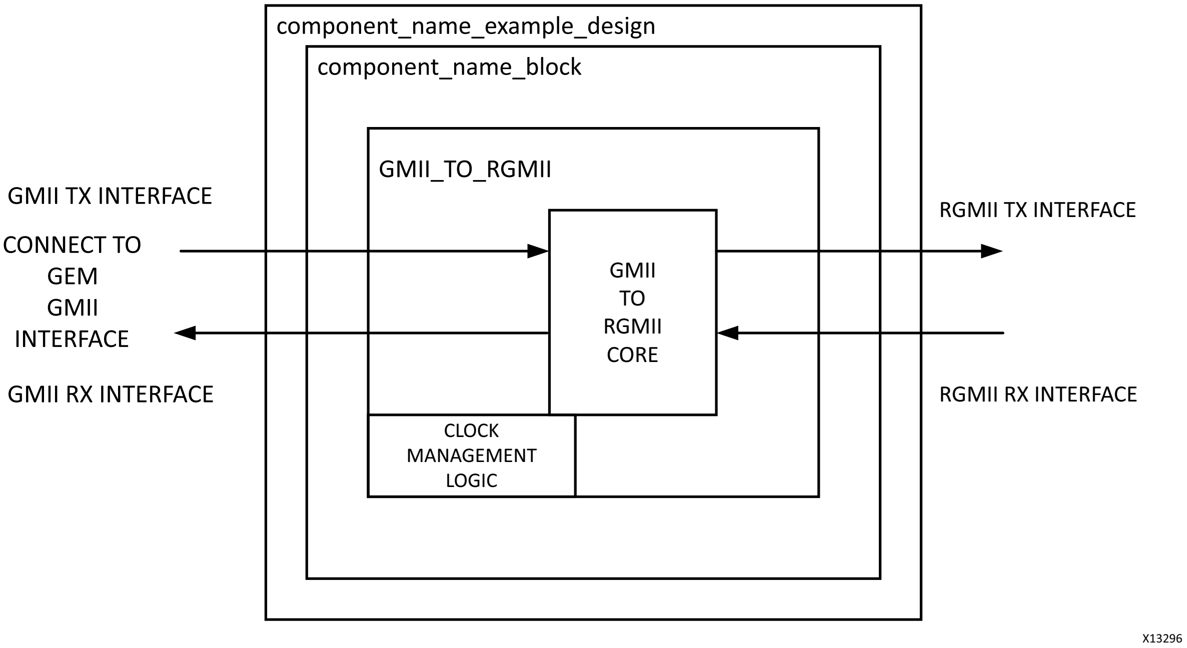


Figure B-2: **Example Design HDL with Internal GMII Clock**

In both examples, the design is split into two hierarchical layers: block level and top level. The block level is designed so that it can be instantiated directly into customer designs and performs the following functions:

- Instantiates the core from HDL
- Connects the physical-side interface of the core to device IOBs, creating an external RGMII.

The top level creates a specific example that can be simulated, synthesized, implemented, and if required, placed on a suitable board and demonstrated in hardware.

Top-Level Example Design HDL

VHDL Vivado® Design Suite:

<project_dir>/*<project_name>*/*<project_name>*.srcs/sources1/ip/*<component_name>*/
<component_name>/example_design/*<component_name>*_example_design.vhd

Vivado Lab Tools

Vivado inserts logic analyzer and virtual I/O cores directly into your design. Vivado Lab Tools allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature represents the functionality in the Vivado IDE that is used for logic debugging and validation of a design running in Xilinx FPGA devices in hardware.

The Vivado logic analyzer is used to interact with the logic debug LogiCORE IP cores, including:

- ILA 2.0 (and later versions)
- VIO 2.0 (and later versions)

Simulation Debug

The simulation debug flow for Questa® SIM is illustrated in Figure B-3. A similar approach can be used with other simulators.

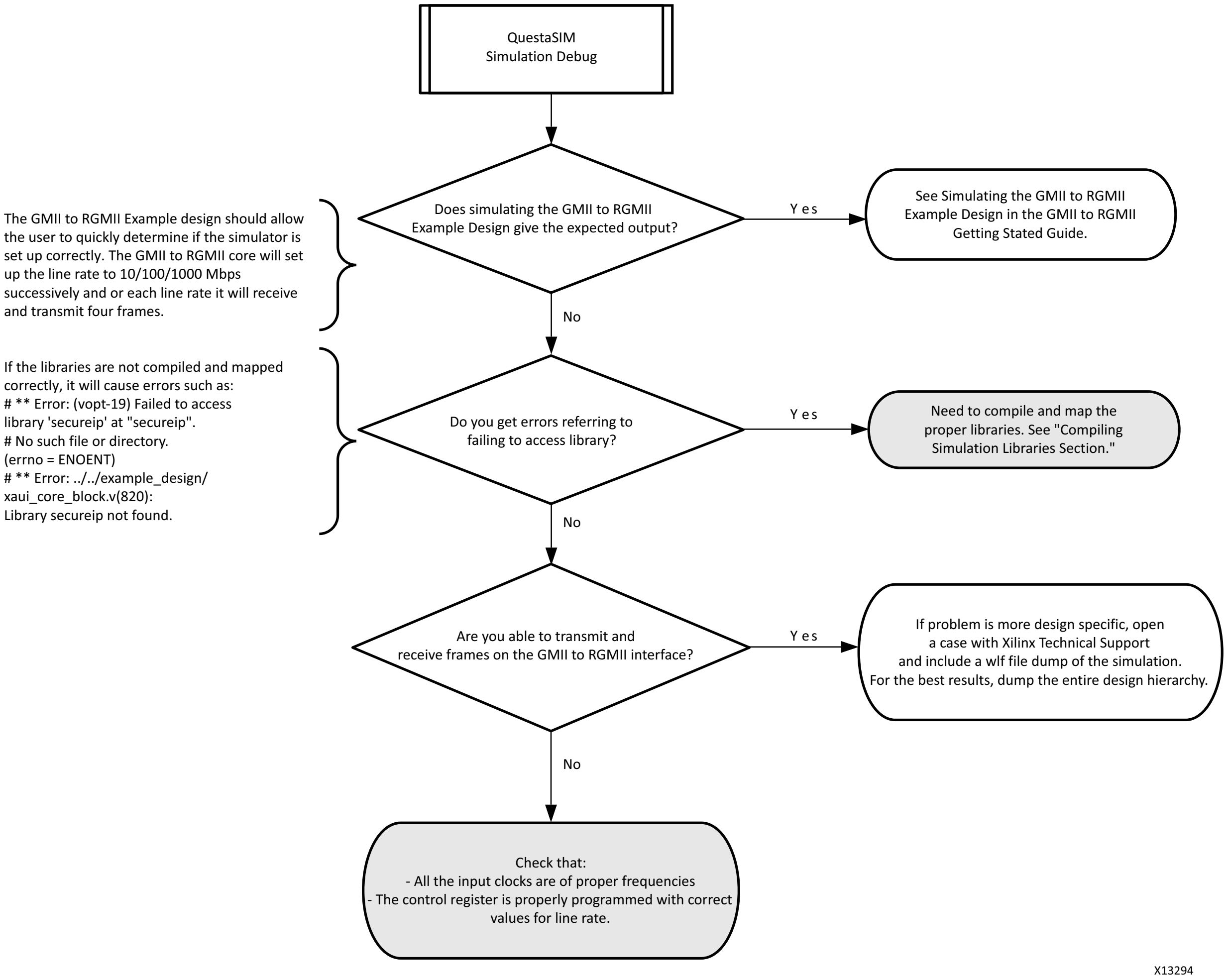

QuestaSIM
Simulation Debug
Does simulating the GMII to RGMII Example Design give the expected output?
Yes
See Simulating the GMII to RGMII Example Design in the GMII to RGMII Getting Stated Guide.
The GMII to RGMII Example design should allow the user to quickly determine if the simulator is set up correctly. The GMII to RGMII core will set up the line rate to 10/100/1000 Mbps successively and or each line rate it will receive and transmit four frames.
No
Do you get errors referring to failing to access library?
Yes
Need to compile and map the proper libraries. See "Compiling Simulation Libraries Section."
If the libraries are not compiled and mapped correctly, it will cause errors such as:
# ** Error: (vopt-19) Failed to access library 'secureip' at "secureip".
# No such file or directory.
(errno = ENOENT)
# ** Error: ../../example_design/ xaui_core_block.v(820):
Library secureip not found.
No
Are you able to transmit and receive frames on the GMII to RGMII interface?
Yes
If problem is more design specific, open a case with Xilinx Technical Support and include a wlf file dump of the simulation. For the best results, dump the entire design hierarchy.
No
Check that:
- All the input clocks are of proper frequencies
- The control register is properly programmed with correct values for line rate.
X13294

Figure B-3: **Simulation Debug Flow**

Hardware Debug

Hardware issues can range from link bring-up to problems seen after hours of testing. This section provides debug steps for common issues. The Vivado Lab Tools are a valuable resource to use in hardware debug. The signal names mentioned in the following individual sections can be probed using the Vivado Lab Tools for debugging the specific problems.

Many of these common issues can also be applied to debugging design simulations. Details are provided on:

- General checks
- Problems with the MDIO
- Problems with data reception or transmission
- Problems with high bit error rates

General Checks

Ensure that all the timing constraints for the core were properly incorporated from the example design and that all constraints were met during implementation.

- Does it work in post-place and route timing simulation? If problems are seen in hardware but not in timing simulation, this could indicate a PCB issue. Ensure that all clock sources are active and clean.
- If using MMCMs in the design, ensure that all MMCMs have obtained lock by monitoring the `LOCKED` port.
- If your outputs go to 0, check your licensing.

Problems with the MDIO

- Ensure that the MDIO is driven properly. See MDIO Management Interface for detailed information about performing MDIO transactions.
- Check that the mdc clock is running and that the frequency is 2.5 MHz or less.
- Check that the PHYAD field placed into the MDIO frame matches the attribute value C_PHYADDR of the core.

Problems with Data Reception or Transmission

When no data is being received or transmitted, ensure that a valid link has been established between the external PHY and its link partner, either by auto-negotiation or manual configuration. After the link has been successfully established, ensure that the driver software configures the core register 0x10 with correct values for speed.

Note: By default, the speed is set to 10 Mb/s.

Problems with High Bit Error Rates

On the RGMII interface, the RX data should provide sufficient set-up time with regard to the RX clock so that it can be sampled correctly. The IDLEAY components are provided on the RX datapath. The correct delay value must be specified so that the RX data can be correctly sampled.

Appendix C

Additional Resources

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see the Xilinx Support website at:

www.xilinx.com/support.

For a glossary of technical terms used in Xilinx documentation, see:

www.xilinx.com/company/terms.htm.

References

These documents provide supplemental material useful with this product guide:

1. *Reduced Gigabit Media Independent Interface (RGMII) Version 2.0*
2. *Vivado Design Suite User Guide - Logic Simulation* (UG900)
3. *Vivado Design Suite User Guide - Implementation* (UG904)
4. Vivado design tools user documentation
5. *Vivado Design Suite User Guide: Designing with IP* (UG896)
6. IEEE Std 802.3-2008
7. *Zynq-7000 All Programmable SoC Technical Reference Manual* (UG585)
8. *Vivado Design Suite Migration Methodology Guide* (UG911)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
03/20/2013	1.0	Initial Xilinx release. Based on PB014.

Notice of Disclaimer

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at http://www.xilinx.com/warranty.htm; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications: http://www.xilinx.com/warranty.htm#critapps.