Preliminary Data Sheet

BCM54285

Octal QSGMII Copper/Fiber Gigabit Ethernet Transceiver

GENERAL DESCRIPTION

The Broadcom® BCM54285 is a fully integrated octal gigabit transceiver with support for Energy Efficient Ethernet™ (EEE), Synchronous Ethernet (SyncE), and IEEE 1588v2 standards.

The MDI twisted-pair transceiver consists of eight triple-speed 10/100/1000BASE-T Ethernet transceivers or eight QSGMII to fiber (100BASE-FX, 1000BASE-X, or SGMII-Slave) interfaces.

When in copper mode, the PHY performs all of the physical layer (PHY) functions for 10BASE-T, 100BASE-TX 1000BASE-T, and on standard Category 5 UTP cable.

When in QSGMII to fiber mode, the PHY performs all of the physical layer functions for 100BASE-FX, 1000BASE-X, and SGMII-Slave.

The BCM54285 is designed to be compliant with the QSGMII industry standards.

The BCM54285 is based on the proven digital-signal processor technology of Broadcom, combining digital adaptive equalizers, ADCs, phase-locked loops (PLL), line drivers, encoders, decoders, echo cancelers, crosstalk cancelers, and all other required support circuitry integrated into a single, monolithic CMOS chip.

Designed for reliable operation over worst-case Category 5 cable plants, the BCM54285 automatically negotiates with any transceiver on the opposite end of the wire to agree on an operating speed. The PHY can also evaluate the condition of the twisted-pair wiring to ensure that the wiring can support operation at gigabit speeds, and detect and correct most common wiring problems. The device continually monitors both the wiring and the opposing transceiver, and alerts the system if it detects potential problems with reliable operation.

APPLICATIONS

- High-density Gigabit Ethernet (GbE) switches and routers.

FEATURES

- QSGMII interface
- Support for IEEE 802.3™-compliant copper line interfaces:
  - 1000BASE-T
  - 100BASE-TX
  - 10BASE-T
- Support for the following fiber line interfaces:
  - 1000BASE-X
  - 100BASE-FX
  - SGMII-Slave
- Integrated twisted-pair termination resistors
- IEEE 802.3az-compliant (EEE)
  - Support for native EEE MACs
  - Support for legacy non-EEE MACs using AutogrEEEn® mode
- Ethernet@WireSpeed™
- Cable plant diagnostics that detect cable plant impairments
- IEEE 1588v2-compliant
  - One-step clock
  - On-chip timestamping
- ITU-T Y.1731 delay measurement support
  - On-chip timestamping
  - One-way and two-way in both directions
- Sync_E support
  - Two recovered clocks
  - Two recovered clock lock outputs
- Jumbo frame support for 16 KB packets
- 3.3V digital I/O
- Support for only two power supplies (1.0V and 3.3V)
- Line-side and switch-side loopbacks
- Dual MDIO support for reduced latency
- Programmable LEDs
- Robust Cable ESD (CESD) tolerance
- Low EMI emissions
- IEEE 1149.1 and 1149.6 (ACJTAG) boundary scan
- Package: 21 mm x 21 mm 400-ball FBGA

54285-DS102-R

5300 California Avenue • Irvine, CA 92617 • Phone: 949-926-5000 • Fax: 949-926-5203

April 12, 2013

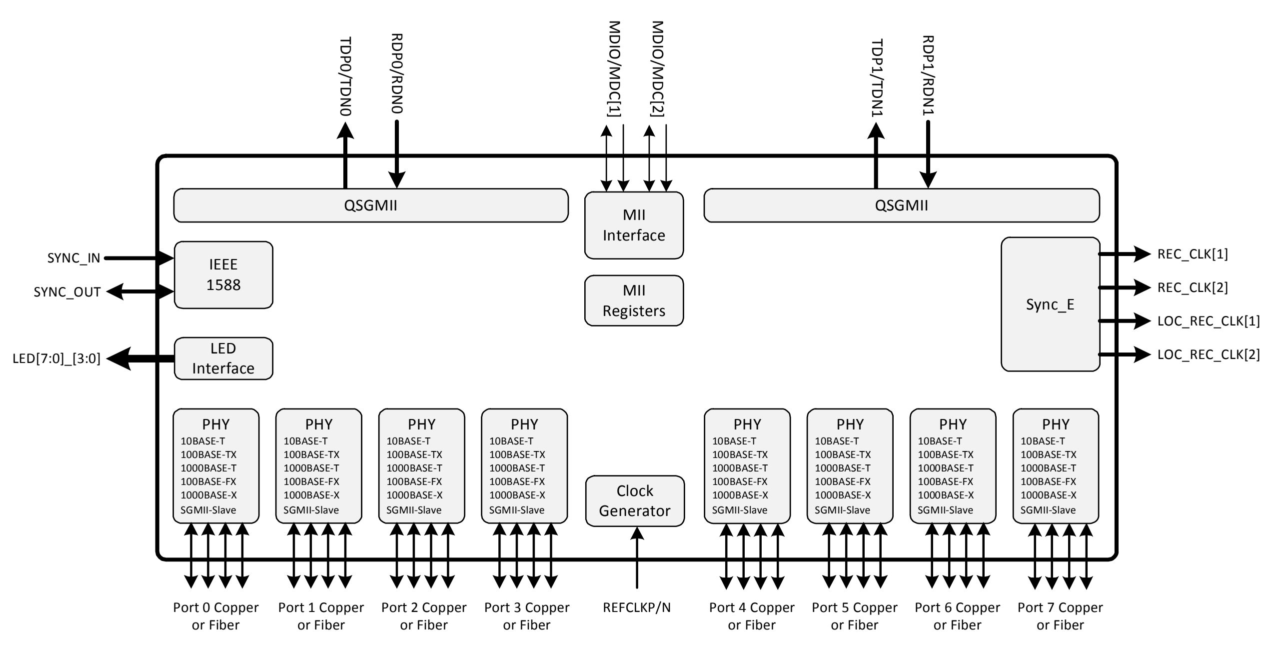


Figure 1: Functional Block Diagram

Revision History

Change Description	***Customer Impact***	***Action Items***
Revision: 54285-DS102-R Date: 04/12/13 ***Note***: Page numbers referred to are valid only for this revision of the document.		
Updated:		
Global change from follow-up to Follow_Up.	None	None
Global change fromIPV4 to: IPv4.	None	None
Global change fromIPV6 to: IPv6.	None	None
Global change from802.1as to 802.1AS.	None	None
Removed LLPD from "AutogrEEEn Flow" on page 23.	None	None
Updated QSGMII description in "QSGMII" on page 37.	None	None
Updated QSGMII description in Table 9: "QSGMII Interface Balls," on page 37.	None	None
Updated "Wiremap and Pair Skew Correction" on page 42.	None	None
Updated Fiber, QSGMII, SCLK_FREQ, TRST, SYNC_IN, and SYNC_OUT descriptions in Table 22: "Ball Descriptions," on page 72.	None	None
Added new ESR row in Table 29: "Crystal Clock Parameters," on page 95.	None	None
Updated Figure 28: "Single-Ended REFCLK Input Timing," on page 108.	None	None
Updated Figure 29: "Differential REFCLK Input Timing," on page 109.	None	None
Updated Table 44: "QSGMII to Copper Current Consumption," on page 114.	None	None
Updated Output differential voltage and definition of Load Type 2 in Table 55: "QSGMII Transmitter," on page 118.	None	None
Updated definition of Load Type 2 in Table 57: "QSGMII Receiver," on page 119.	None	None
Added:		
Added "Fiber Interface" on page 49.	None	None
Added Table 45: "QSGMII to Fiber Current Consumption," on page 115.	None	None
Revision: 54285-DS101-R Date: 11/01/12 ***Note***: Page numbers referred to are valid only for this revision of the document.		
Updated:		
Global change from MDI MII to MDIO MII.	None	None

Change Description	***Customer Impact***	***Action Items***
Global change from Differential Clock to CML Differential Clock.	None	None
Figure 10: “One Port of the PHY with IEEE 1588 Support,” on page 33.	None	None
Added REFCLKP/REFCLKN have an internal 100 ohm differential impedance in“CML Differential Mode” on page 122.	None	None
Changed TBD to 1 ns in Table 50: “REFCLK Input Timing,” on page 139 and Table 51: “REFCLKP/N Clock Input Timing,” on page 140.	None	None
Changed MDIO output delay from 15 ns to 50 ns for OVDDMDIO = 1.2V in Table 52: “Management Interface Timing,” on page 141.	None	None
Changed min. and max. TCK to TDO Delay from 5.0 ns to 4.5 ns and 15 ns to 20 ns in Table 54: “JTAG Timing,” on page 142.	None	None
Changed vil from 0.8V to 0.7V and voh from OVDD-0.4V to OVDD - 0.45V in Table 64: “JTAG: OVDDJTAG Operating at 2.5V,” on page 147.	None	None
Changed Reference clock input voltage swing differential from 1600 to 2000 in Table 67: “CML Differential Reference Clock,” on page 148.	None	None
Changed Differential Input Impedance from 80 to 90 and from 120 to 130 in Table 67: “CML Differential Reference Clock,” on page 148.	None	None
Changed Reference clock input voltage swing differential from 500 to 600 in Table 67: “CML Differential Reference Clock,” on page 148.	None	None
Changed C0 to C1 in Table 77: “Ordering Information,” on page 158.	C0 silicon is no longer supported.	Order C1 silicon.
Added:		
Added register writes for QSGMII LPI enable in“Native EEE Flow” on page 26.	None	None
Added PU to MDIO I/O type and PD to MDC I/O type in Table 31: “Ball Descriptions,” on page 86.	None	None
Added “1000BASE-X Line-Side Loopback” on page 132.	None	None
Revision: 54285-DS100-R Date: 04/25/12 ***Note***: Page numbers referred to are valid only for this revision of the document.		
Initial release.	–	–

Broadcom Corporation
5300 California Avenue
Irvine, CA 92617

Table of Contents

List of Figures

List of Tables

About This Document

Purpose and Audience

This document provides details of the functional, operational, and electrical characteristics of the BCM54285. It is intended for hardware design, application, and OEM engineers.

Acronyms and Abbreviations

In most cases, acronyms and abbreviations are defined on first use.

Acronyms and abbreviations in this document are also defined in Appendix A: "Acronyms and Abbreviations," on page 126.

For a comprehensive list of acronyms and other terms used in Broadcom documents, go to: http://www.broadcom.com/press/glossary.php.

Document Conventions

The following conventions may be used in this document:

Convention	***Description***
Bold	User input and actions: for example, type **exit**, click **OK,** press **Alt+C**
`Monospace`	Code: `#include <iostream>` HTML: `<td rowspan = 3>` Command line commands and parameters: `wl [-l] <command>`
< >	Placeholders for *required* elements: enter your <username> or `wl <command>`
[ ]	Indicates *optional* command-line parameters: `wl [-l]` Indicates bit and byte ranges (inclusive): [0:3] or [7:0]

The following notational conventions are used in this document:

- A bar over a signal name indicates that it is active low. For example, $\overline{\text{RESET}}$.
- In register descriptions:
  - Bit[n] indicates bit n. For example, Bit[7] indicates bit 7.
  - Bits[n:m] indicates a range from bit n to bit m. For example, Bits[7:0] indicates bits 7 through 0, inclusive.
- Hexadecimal numbers can be represented by the use of leading 0x. For example, 0x13B.
- Binary numbers can be represented by the use of leading 0'b. For example, 1'b1, 2'b01, 3'b101.

References

The references in this section may be used in conjunction with this document.

Note: Broadcom provides customer access to technical documentation and software through its Customer Support Portal (CSP) and Downloads & Support site (see Technical Support).

For Broadcom documents, replace the “xx” in the document number with the largest number available in the repository to ensure that you have the most current version of the document.

Document (or Item) Name	*Number*	*Source*
Broadcom Items		
[1] Synchronizing Broadcom PHYs for Point-to-Point Synchronous Ethernet Applications, Application Note	SyncE-AN1xx-R	Broadcom CSP
[2] Reflow Process Guidelines for Surface Mount Assemblies, Application Note	PACKAGING-AN1xx-R	Broadcom CSP

Technical Support

Broadcom provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates through its customer support portal (https://support.broadcom.com). For a CSP account, contact your Sales or Engineering support representative.

In addition, Broadcom provides other product support through its Downloads & Support site (http://www.broadcom.com/support/).

Section 1: Introduction

Overview

This document provides the following information on the BCM54285 PHY:

Section 2: Functional Description

Overview

The BCM54285 is an octal Gigabit Ethernet transceiver that supports interfaces and features shown in Table 1.

Table 1: Auto-Negotiation Register

Switch Interface	***MDI Interface***	***Protocol for MDI***	***Additional Features***
QSGMII	Copper	• 1000BASE-T • 100BASE-TX • 10BASE-T	• Energy Efficient Ethernet: IEEE 802.3az-compliant • AutogrEEEn technology • 1588: IEEE 1588v2-compliant • Synchronous Ethernet ***Note:*** See Figure 2.
QSGMII	Fiber or SGMII-Slave	• 1000BASE-X • 100BASE-FX • SGMII-Slave	• 1588: IEEE 1588v2-compliant • Synchronous Ethernet ***Note:*** See Figure 3: "QSGMII-to-Fiber Block Diagram," on page 18.

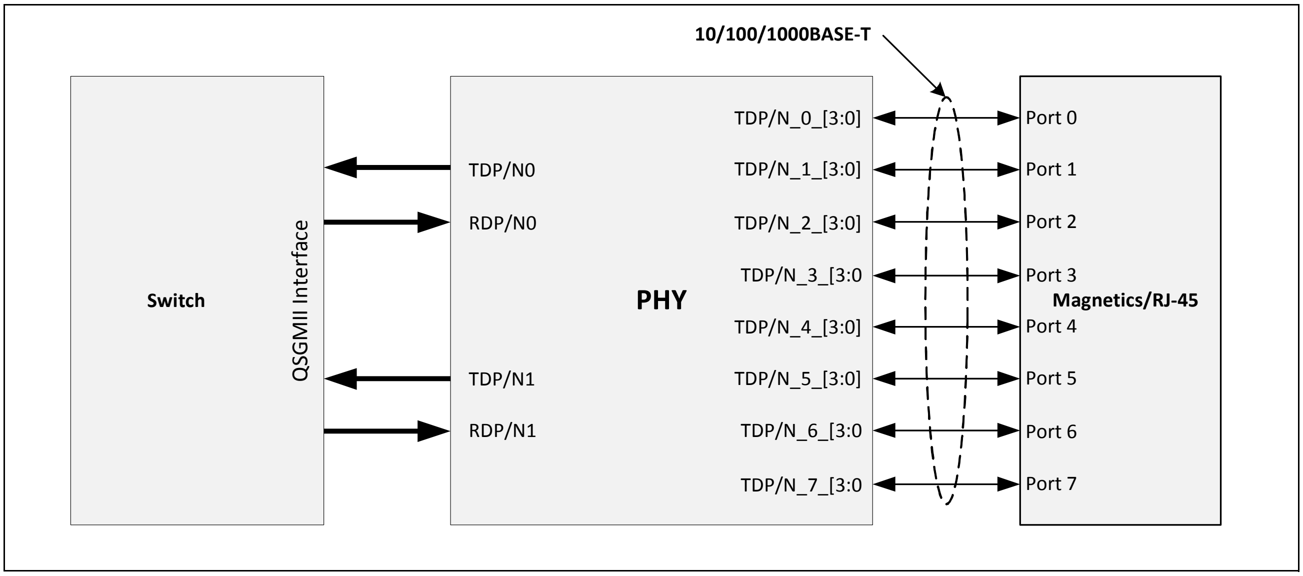


Figure 2: QSGMII-to-Copper Block Diagram

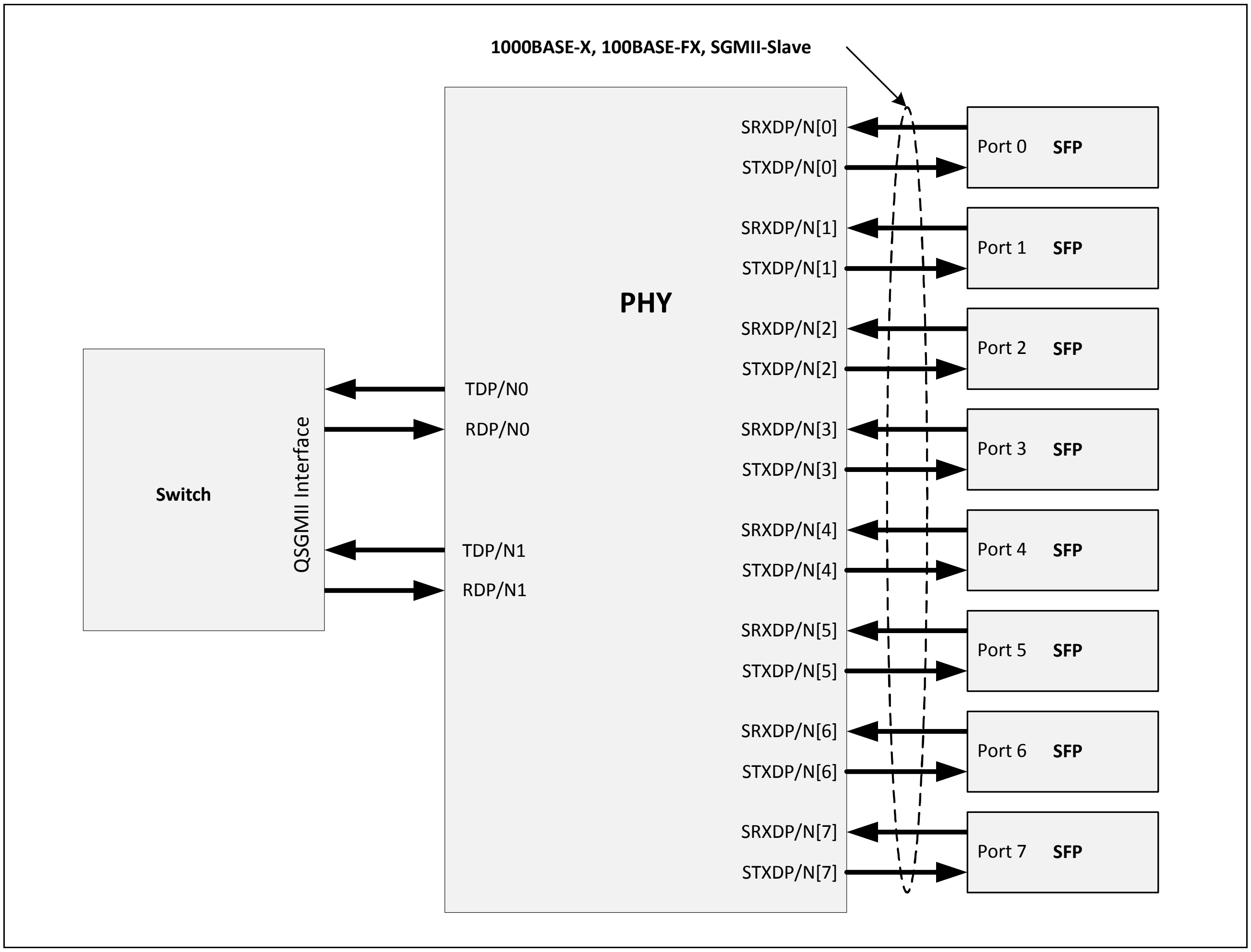


Figure 3: QSGMII-to-Fiber Block Diagram

Section 3: Key Features

Overview

This section describes the key features of the BCM54285.

- Energy Efficient Ethernet (EEE): IEEE 802.3az-compliant
- AutogrEEEn
- 1588: IEEE 1588v2-compliant
- Synchronous Ethernet

Energy Efficient Ethernet

The BCM54285 contains support for Energy Efficient Ethernet. EEE is IEEE 802.3az, an extension of the IEEE 802.3 standard. EEE defines support for the PHY to operate in Low-Power Idle (LPI) mode, which when enabled supports quiet times during low link utilization allowing both sides of a link to disable portions of each PHY’s operating circuitry and save power.

The BCM54285 offers two basic modes of operation: Native EEE mode for switches that support LPI signaling across the QSGMII interface, and AutogrEEEn mode for legacy switches that do not support LPI signaling across the QSGMII interface. When in either Native or AutogrEEEn mode, the PHY supports the following:

- Support for 100BASE-TX (auto-negotiation must be enabled).
- Support for 1000BASE-T (auto-negotiation must be enabled).
- Link status does not change.
- Frames are not dropped or corrupted.
- The transition time to and from the lower power levels is transparent to upper layer protocols and applications.

In addition to the standard EEE operation, the BCM54285 supports the following enhancements to the EEE functions:

- Native mode enhancements:
  - QSGMII auto-negotiation link partner register includes new EEE Capability bit (bit[9] = 1’b1: LPI Capability).
- AutogrEEEn mode enhancements:
  - Fixed latency
  - Variable latency

Native EEE Mode

Native EEE mode is used for switches that support IEEE 802.3az LPI code group signaling through the QSGMII interfaces. Figure 4 shows LPI support in the switch.

The switch also has to be aware that the medium is not immediately available when in LPI mode. This means the switch needs to be able to do the following:

- Hold off data transmission when the link is in LPI mode.
- Account for wake-up time when transitioning from LPI to the Active state when it wants to initiate data transmission.
- Optionally support the IEEE 802.3AB protocol (LLDP). This is used by devices to negotiate additional wake-up times.

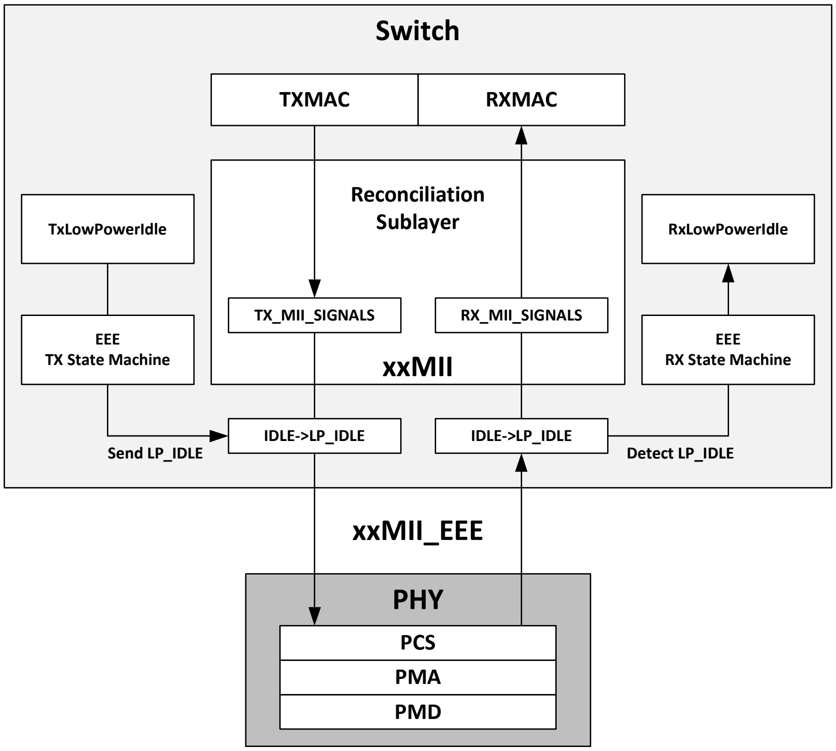


Figure 4: Native EEE Mode (Switch Supported EEE LPI)

Native EEE Flow

The following steps are needed to enable native EEE mode on a per port basis:

1. Enable native EEE mode.
   a. Write Clause 45, DEVAD 0x7, address 0x803D, bits[15:14] = 2'b11 (Default setting).
2. Advertise normal auto-negotiation capabilities.
   a. Write register 0x04 with 10BASE-T and 100BASE-TX abilities.

   b. Write register 0x09 with 1000BASE-T abilities.

3. Advertise EEE auto-negotiation capabilities.

   a. Write Clause 45 DEVAD 0x7, address 0x3C, bit[1] = 1’b1 (advertise 100BASE-TX EEE ability).

   b. Write Clause 45 DEVAD 0x7, address 0x3C, bit[2] = 1’b1 (advertise 1000BASE-T EEE ability).

4. Initiate auto-negotiation.

   a. Write register 0x0, bit[12] = 1’b1 (restart auto-negotiation).

   Local and remote PHY auto-negotiate speed, duplex, remote fault, pause, master/slave, next page, and EEE abilities.

   The link is established (10BASE-T, 100BASE-TX, or 1000BASE-T).

   The switch looks at the EEE_RESOLUTION_STATUS register, Clause 45 DEVAD 0x7, address 0x803E, bits[2:1] to determine what EEE speeds are supported. Bit[2] = 1’b1: Both local device and link partner advertise EEE 1000BASE-T capability. Bit[1] = 1’b1: Local and link partner advertise EEE 100BASE-TX mode.

5. Read EEE 1000BASE-T resolution.

   a. Clause 45, DEVAD 0x7, address 0x803E, bit[2] = 1’b1 (EEE_1000T_Resolution).

6. Read EEE 100BASE-TX resolution.

   a. Clause 45, DEVAD 0x7, address 0x803E, bit[1] = 1’b1 (EEE_100TX_Resolution).

If both the local and remote PHYs support EEE, they can optionally negotiate longer system wake-up times (T_{w_sys}) through LLPD. The following default settings are used if LLPD is not used:

- T_{w_sys} default for 100BASE-TX = 30 µs
- T_{w_sys} default for 1000BASE-T = 16.5 µs

The switch determines when the BCM54285 transmits LPI signals on the MDI. The BCM54285 transmits LPI signals when the BCM54285 receives the newly defined code groups from the switch as specified in IEEE 802.3az.

- If a 100BASE-TX link is established, the BCM54285 can start sending LPI.
- If a 1000BASE-T link is established, the BCM54285 only enters LPI mode after it transmits LP_Sleep and receives LP_Sleep from the remote PHY.

The BCM54285 communicates and coordinates the LPI transition across the MDI. See “MDI LPI Operation” on page 24. The BCM54285 keeps the link-related parameters up to date during refresh. The switch determines when the BCM54285 will stop sending LPI signals on the MDI and transition to normal mode.

The BCM54285 stops transmitting LPI signals when the BCM54285 receives normal IPGs from the switch to stop transmitting LPI signals. After the Tw_PHY time, the switch may send normal data.

Note: The user is not allowed to send normal data to terminate LPI request.

LPI Mode Status

To determine when the PHY is receiving LP Idle from the switch, read PCS_STATUS_1 register.

- Clause 45, DEVAD 0x3, address 0x1, bit[9] = 1'b1 (transmit PCS is currently receiving LP idle)
- Clause 45, DEVAD 0x3, address 0x1, bit[8] = 1'b1 (receive PCS is currently receiving LP idle)

AutogrEEEn Mode

AutogrEEEn mode is used for legacy switches that do not support IEEE 802.3az LPI code group signaling through the QSGMII interfaces. Figure 5 shows LPI support in the PHY. AutogrEEEn mode transmits the same LPI signals on the MDI as Native EEE mode, making it compatible with PHYs that support either Native EEE or AutogrEEEn mode.

AutogrEEEn mode has the following benefits:

- The switch does not need to be aware that the PHY is transmitting LPIs.
- The switch assumes that the media is always available to it.
- Allows legacy switches to connect to either EEE enabled link partners or non-EEE enabled link partners.

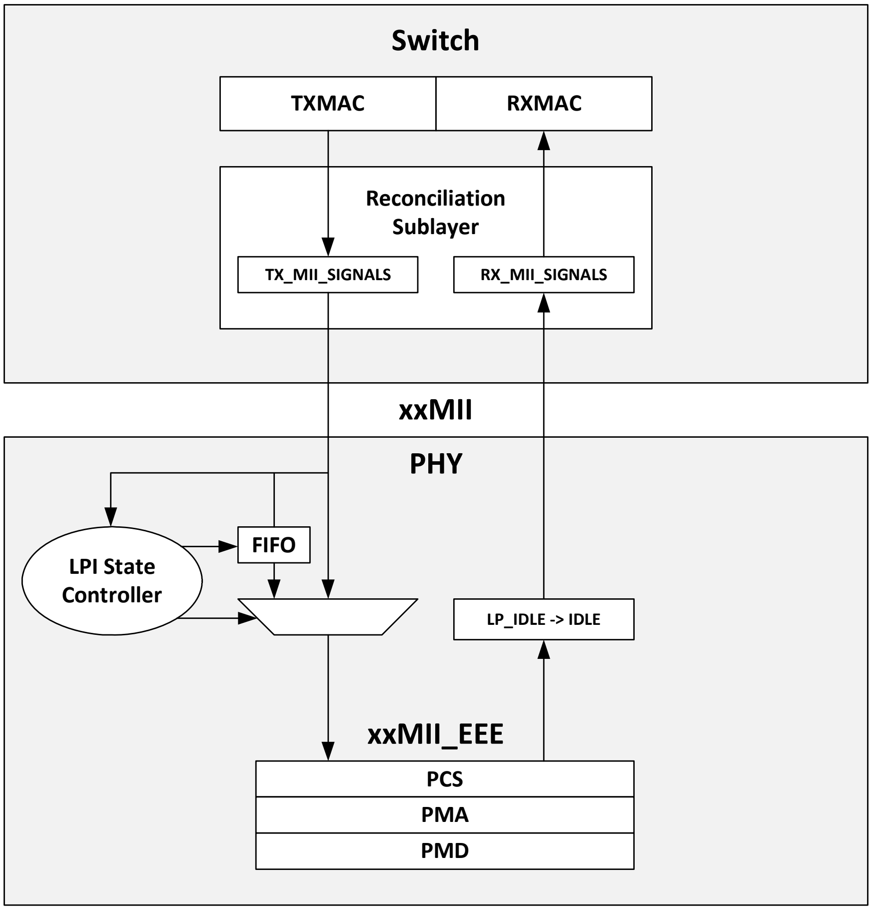


Figure 5: AutogrEEEn Mode (PHY Supported EEE LPI)

AutogrEEEn Flow

The following are the steps needed to enable AutogrEEEn mode on a per port basis. AutogrEEEn mode is disabled by default.

1. Enable EEE mode.
    a. Write Clause 45, DEVAD 0x7, address 0x803D, bits[15:14] = 2’b11 (default setting).
2. Advertise EEE auto-negotiation capabilities.
    a. Write Clause 45, DEVAD 0x7, address 0x3C, bit[1] = 1’b1 (100BASE-TX EEE ability).
    b. Write Clause 45, DEVAD 0x7, address 0x3C, bit[2] = 1’b1 (1000BASE-T EEE ability).
3. Advertise normal auto-negotiation capabilities.
    a. Write register 0x04 with 10BASE-T and 100BASE-TX abilities.
    b. Write register 0x09 with 1000BASE-T abilities.
4. Enable optional variable/fixed latency modes. See “Variable/Fixed Latency Modes” on page 25.
    - RDB_Register, offset 0x800, bit[2] = 1’b1 (enable variable latency mode) or
    - RDB_Register, offset 0x800, bit[2] = 1’b0 (enable fixed latency mode)
5. Initiate auto-negotiation.
    a. Write register 0x00, bit[9] = 1’b1 (restart auto-negotiation). Local and remote PHY auto-negotiate speed, duplex, remote fault, pause, master/slave, next page, and EEE abilities. The link is established (10BASE-T, 100BASE-TX, or 1000BASE-T).

       The switch looks at the EEE_RESOLUTION_STATUS register, Clause 45 DEVAD 0x7,address 0x803E, bits[2:1] to determine what EEE speeds are supported. Bit[2] = 1’b1: Both local device and link partner advertise EEE 1000BASE-T capability. Bit[1] = 1’b1: Local and link partner advertise EEE 100BASE-TX mode.
6. Read EEE 1000BASE-T resolution.
    a. Read Clause 45, DEVAD 0x7, address 0x803E, bit[2] = 1’b1 (EEE_1000T_Resolution).
7. Read EEE 100BASE-TX resolution.
    a. Read Clause 45, DEVAD 0x7, address 0x803E, bit[1] = 1’b1 (EEE_100TX_Resolution:).
8. Enable AutogrEEEn on port-by-port basis:
    a. Port 0: Write RDB_Register, offset 0x8000, bit[0] = 1’b1
    b. Port 1: Write RDB_Register, offset 0x8002, bit[0] = 1’b1
    c. Port 2: Write RDB_Register, offset 0x8004, bit[0] = 1’b1
    d. Port 3: Write RDB_Register, offset 0x8006, bit[0] = 1’b1
    e. Port 4: Write RDB_Register, offset 0x8008, bit[0] = 1’b1
    f. Port 5: Write RDB_Register, offset 0x800A, bit[0] = 1’b1
    g. Port 6: Write RDB_Register, offset 0x800C, bit[0] = 1’b1
    h. Port 7: Write RDB_Register, offset 0x800E, bit[0] = 1’b1

The BCM54285 transmits LPI signals on the MDI when the number of consecutive idle symbols is equal to the EEE_Idle threshold.

- If a 100BASE-TX link is established, the BCM54285 can start sending LPI.

- If a 1000BASE-T link is established, the BCM54285 will only enter LPI mode after it transmits LP_Sleep and receives LP_Sleep from the remote PHY.

The BCM54285 communicates and coordinates the LPI transition across the MDI (see "MDI LPI Operation" on page 24). The BCM54285 keeps the link related parameters up to date through refresh. The BCM54285 will stop sending LPI signals on the MDI and transition to normal mode when a transmit data is received from the switch.

QSGMII Auto-Negotiation

QSGMII auto-negotiation register in addition to the link, duplex, and speed bits now contains a EEE capable bit (bit[9]) to indicate if the remote device is EEE capable as shown in Table 2.

Table 2: Auto-Negotiation Register

Bit	Name	Description
15	Copper link	1'b1 = Link up. 1'b0 = Link down.
14	Reserved	Reserve for auto-negotiation acknowledge as specified in IEEE 802.3z.
13	Reserved	Reserved for future use.
12	Copper duplex	1'b1 = Full-duplex mode. 1'b0 = Half-duplex mode.
11:10	Copper speed	2'b11 = Reserved. 2'b10 = 1000BASE-T. 2'b01 = 100BASE-TX. 2'b00 = 10BASE-T.
9	EEE capable	1'b1 = Remote device is EEE capable. 1'b0 = Remote device is not EEE capable.
8:1	Reserved	Write as 1'b0. Reserved for future use.
0	QSGMII selector	Write as 1'b1.

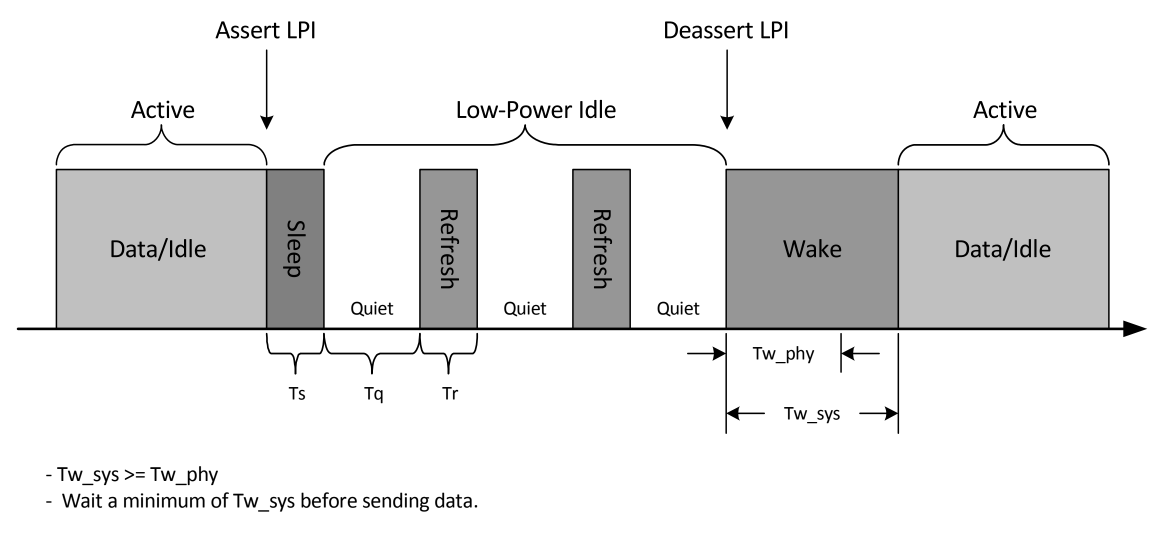
MDI LPI Operation

In 100BASE-TX mode, the local PHY transmits a special LP_Sleep signal to communicate to the link partner that the local system is entering LPI mode. In 1000BASE-T mode, the transmit function of the local PHY enters a quiet mode only after the local PHY transmits LP_Sleep and receives LP_Sleep from the remote PHY. If the remote PHY does not signal LPI, then neither PHY can go quiet. The LPI requests are still passed from one end to the other end of the link, since other system energy savings may be achieved even if the PHY link does not go quiet.

There are three states visible above the EEE PHY when in native EEE mode. These states are shown below and in Figure 6 on page 25. For AutogrEEEn mode these states are handled by the BCM54285.

- Active—Transmission of data and normal idle.
- Low-power idle—Transmission of low-power idle.
- Hold—Transition state between low-power idle and active.

During LPI mode, only two states are visible; sleep and quiet. For times in each state and values refer to IEEE 802.3az Clause 24 table 24-2 for 100BASE-TX and Clause 40.12.5 for 1000BASE-T.



- Tw_sys >= Tw_phy
- Wait a minimum of Tw_sys before sending data.

Figure 6: AutogrEEEn Mode (PHY Supported EEE LPI)

Variable/Fixed Latency Modes

When in AutogrEEEn mode, the BCM54285 can support either variable or fixed latency modes. When operating in variable latency mode:

- Non-IDLE characters are written into the FIFO.
- The FIFO will drain as the IDLEs accumulate in the traffic profile.
- The FIFO will be bypassed once it is empty.

When operating in fixed latency mode (RDB_Register, offset 0x800, bit[2] = 1’b0), IDLE characters are stuffed in the FIFO to maintain a constant latency value in the data stream.

IEEE 1588 Support

The BCM54285 supports IEEE 1588v2 Transparent Clock (TC) operation. The BCM54285 can be used as the PHY in a system that supports either one-step operation or two-step TC operation where Follow_Up messages are sent. A summary of modes supported by the BCM54285 is as follows:

- One-step or two-step TC support.
- Grand master or slave clock support.
- Various synchronization mode support.
- IEEE frame formats support.
- Fixed or variable latency support.
- IEEE 1588 over MPLS. Up to three labels of support.

- Egress timestamp captured in FIFO on a per port basis.
- Egress timestamp PTP option available for both 48-bits and 64-bits.

One-Step Clock

A one-step clock provides time information using a single event message, which means that Follow_Up messages are not sent. Special hardware is required to embed the timestamps in the out-going packets. The BCM54285 has the internal hardware to update the timestamps. Timestamping is done as close as possible to the MDI, ensuring the highest accuracy. The PHY can update Correction Fields (CF) included in IEEE 1588 packet messages. The PHY supports updating CF on IEEE 802.1AS, IEEE 1588 L2, IEEE 1588 IPv4/UDP, and IEEE 1588 IPv6/UDP packets. Figure 7 shows a block diagram of one port of the PHY showing ingress and egress parsers to parse the packets, as well as the Network Synchronization Engine (NSE) which handles external synchronization and updating the CF.

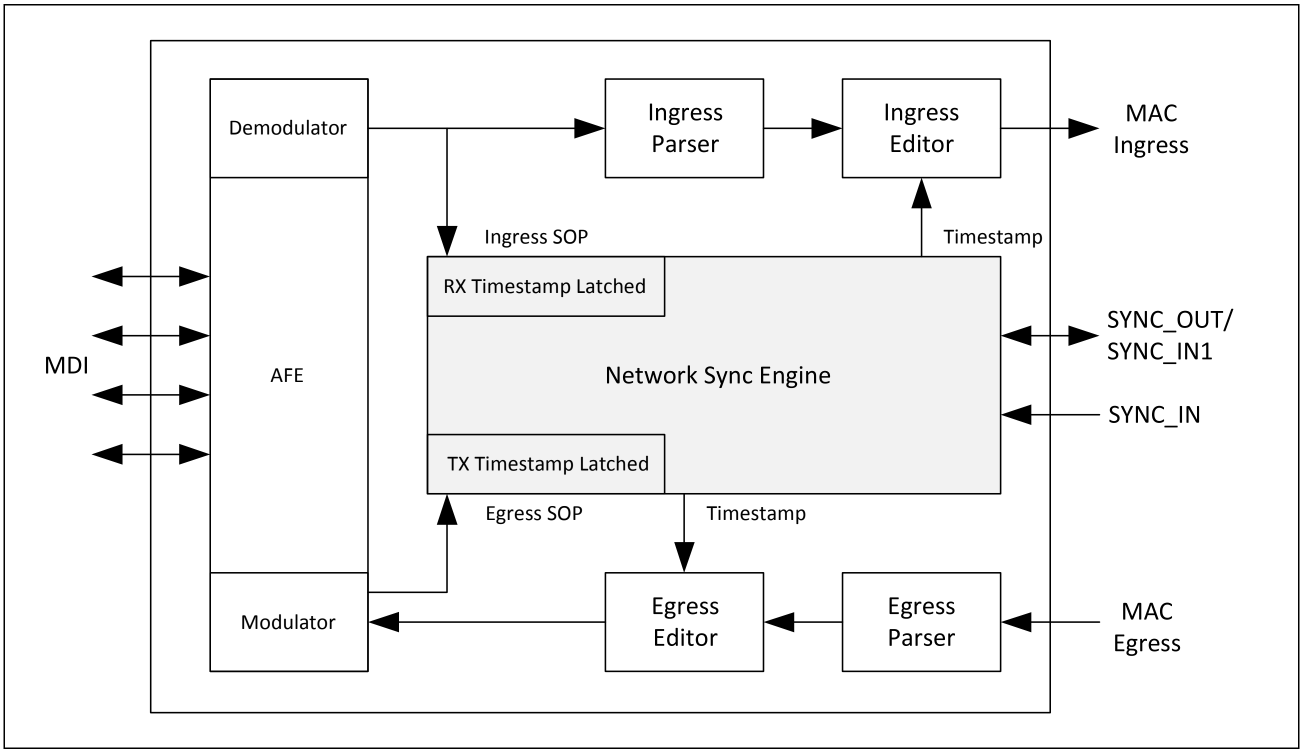


Figure 7: One Port of the PHY with IEEE 1588 Support

Transparent Clock Support

The BCM54285 features the following Transparent Clock (TC) support:

- MC/UC L2 and L4 packet parsers to identify and timestamp IEEE 1588 event messages.
- MC/UC L2 and L4 packet modification with CRC and UDP checksum update.
- Residence time corrected by subtracting ingress time and adding egress time.
- Timestamp counters that lock to external sync signal, for synchronizing multiple PHYs.

Grand Master Support

The BCM54285 features the following Grand Master (GM) support:

- Support for CPU and non-CPU GM scenarios.
- MC/UC L2/L4 parsers and counters to update timestamp and correction fields.

Slave Clock Support

The BCM54285 features the following Slave Clock (SC) support:

- MC/UC L2 and L4 packet parsers to identify and timestamp IEEE 1588 event messages.
- Timestamp counters that lock to an external sync signal for synchronizing multiple PHYs.
- Access to raw timestamps using MDIO.

Synchronization Support

Synchronization allows the PHY to accept a synchronization signal from a CPU or backplane. The purpose of the synchronization signal is to allow the PHY or subsequent PHYs to lock their timestamp counters. The PHY can support propagation of a synchronization signal from the backplane to other PHYs on the board, simplifying board layout using SYNC_OUT. The PHY can also generate a synchronization pulse when a local timer reaches a certain value.

Synchronization support includes the following features:

- An on-chip Digital PLL (DPLL) locks timestamp counters to the external sync signal.
- The SYNC_IN accepts external synchronization signals.
- The SYNC_OUT generates or accepts synchronization signals. This ball also functions as SYNC_IN1. The default for this ball on power-up is SYNC_IN1.
- The BCM54285 can synchronize other Broadcom PHYs with external synchronization support.

IEEE 1588 Frame Formats

The BCM54285 supports parsing of two basic frame formats: IEEE 802.1AS and IEEE 1588. Frame formats are enabled in the IEEE_1588_RECEIVE_CONTROL register (Expansion D, offset 0x33). IEEE 802.3 SNAP and LLC formats are not supported by the BCM54285 parser. The PHY supports VLAN tags in the frame format. The following tables show the packet formats that can be parsed by the BCM54285:

- Table 3: “IEEE 802.1AS Ethernet II Packet Formats,” on page 28
- Table 4: “IEEE 1588 L2 Packet Formats,” on page 29
- Table 5: “IEEE 1588 IPv4/UDP Packet Formats,” on page 30
- Table 6: “IEEE 1588 IPv6/UDP Packet Formats,” on page 31

Table 3: IEEE 802.1AS Ethernet II Packet Formats

Ethernet II Untagged			***Ethernet II One Tag***			***Ethernet II Two Tags***		
Bytes	***Field***	***Value***	***Bytes***	***Field***	***Value***	***Bytes***	***Field***	***Value***
–	Preamble	55	–	Preamble	55	–	Preamble	55
1	SFD	5D	1	SFD	5D	1	SFD	5D
6	MAC DA	–	6	MAC DA	–	6	MAC DA	–
6	MAC SA	–	6	MAC SA	–	6	MAC SA	–
2	Ethernet type	88F7	2	TPID	8100	2	OTPID	9100
1	Msg type	< 4	2	VLAN	–	2	OVLAN	–
1	PTP version	2	2	Ethernet type	88F7	2	ITPID	–
6	PTP other	–	1	Msg type	< 4	2	IVLAN	–
8	Correction	–	1	PTP version	2	2	Ethernet type	88F7
14	PTP other	–	6	PTP other	–	1	Msg type	< 4
2	Sequence number	–	8	Correction	–	1	PTP version	2
2	PTP other	–	14	PTP other	–	6	PTP other	–
8	TS insertion[a]	–	2	Sequence number	–	8	Correction	–
10	Origin timestamp	–	2	PTP other	–	14	PTP other	–
–	–	–	8	TS insertion[a]	–	2	Sequence number	–
–	–	–	10	Origin timestamp	–	2	PTP other	–
–	–	–	–	–	–	8	TS insertion[a]	–
–	–	–	–	–	–	10	Origin timestamp	–

a. Timestamp insertion can only be used in RX slave clock and specific grand master modes.

Table 4: IEEE 1588 L2 Packet Formats

IEEE1588 L2 Untagged			***IEEE1588 L2 One Tag***			***IEEE 1588 L2 Two Tags***		
Bytes	***Field***	***Value***	***Bytes***	***Field***	***Value***	***Bytes***	***Field***	***Value***
–	Preamble	55	–	Preamble	55	–	Preamble	55
1	SFD	5D	1	SFD	5D	1	SFD	5D
6	MAC DA	–	6	MAC DA	–	6	MAC DA	–
6	MAC SA	–	6	MAC SA	–	6	MAC SA	–
2	Ethernet type	88F7	2	VLAN	–	2	OVLAN	–
1	Msg type	< 4	2	Ethernet type	88F7	2	IVLAN	–
1	PTP version	2	1	Msg type	< 4	2	Ethernet type	88F7
6	PTP other	–	1	PTP version	2	1	Msg type	< 4
8	Correction	–	6	PTP other	–	1	PTP version	2
14	PTP other	–	8	Correction	–	6	PTP other	–
2	Sequence number	–	14	PTP other	–	8	Correction	–
2	PTP other	–	2	Sequence number	–	14	PTP other	–
8	TS insertion[a]	–	2	PTP other	–	2	Sequence number	–
10	Origin timestamp	–	8	TS insertion[a]	–	2	PTP other	–
–	–	–	10	Origin timestamp	–	8	TS insertion[a]	–
–	–	–	–	–	–	10	Origin timestamp	–

a. Timestamp insertion can only be used in RX slave clock and specific grand master modes.

Table 5: IEEE 1588 IPv4/UDP Packet Formats

IPv4/UDP Untagged			IPv4/UDP One Tag			IPv4/UDP Two Tags		
Bytes	**Field**	**Value**	**Bytes**	**Field**	**Value**	**Bytes**	**Field**	**Value**
–	Preamble	55	–	Preamble	55	–	Preamble	55
1	SFD	5D	1	SFD	5D	1	SFD	5D
6	MAC DA	–	6	MAC DA	–	6	MAC DA	–
6	MAC SA	–	6	MAC SA	–	6	MAC SA	–
2	Ethernet type	0800	4	VLAN	–	2	OVLAN	–
20	IPv4 header	–	2	Ethernet type	0800	4	IVLAN	–
0	Options (NA))	–	20	IPv4 header	–	2	Ethernet type	0800
2	UDP source port	–	0	Options (NA)	–	20	IPv4 header	–
2	UDP destination port	319	2	UDP source port	–	0	Options (NA)	–
4	Length checksum	–	2	UDP destination port	319	2	UDP source port	–
1	Msg type	< 4	4	Length checksum	–	2	UDP destination port	319
1	PTP version	2	1	Msg type	< 4	4	Length checksum	
6	PTP other	–	1	PTP version	2	1	Msg type	< 4
8	Correction	–	6	PTP other	–	1	PTP version	2
14	PTP other	–	8	Correction	–	6	PTP other	–
2	Sequence number	–	14	PTP other	–	8	Correction	–
2	PTP other	–	2	Sequence number	–	14	PTP other	–
8	TS insertion[a]	–	2	PTP other	–	2	Sequence number	–
10	Origin timestamp	–	8	TS insertion[a]	–	2	PTP other	–
–	–	–	10	Origin timestamp	–	8	TS insertion[a]	–
–	–	–	–	–	–	10	Origin timestamp	–

a. Timestamp insertion can only be used in RX slave clock and specific grand master modes.

Table 6: IEEE 1588 IPv6/UDP Packet Formats

IPv6/UDP Untagged			*IPv6/UDP One Tag*			*IPv6/UDP Two Tags*		
Bytes	***Field***	***Value***	***Bytes***	***Field***	***Value***	***Bytes***	***Field***	***Value***
–	Preamble	55		Preamble	55	–	Preamble	55
1	SFD	5D	1	SFD	5D	1	SFD	5D
6	MAC DA	–	6	MAC DA	–	6	MAC DA	–
6	MAC SA	–	6	MAC SA	–	6	MAC SA	–
2	Ethernet type	86DD	4	VLAN	–	2	OVLAN	–
40	IPv6 header	–	2	Ethernet type	86DD	4	IVLAN	–
0	Ext. header (NA)	–	40	IPv6 header	–	2	Ethernet type	86DD
2	UDP source port	–	0	Ext. header (NA)	–	40	IPv6 header	–
2	UDP destination port	319	2	UDP source port	–	0	Ext. header (NA)	–
4	Length checksum	–	2	UDP destination port	319	2	UDP source port	–
1	Msg type	< 4	4	Length checksum		2	UDP destination port	319
1	PTP version	2	1	Msg type	< 4	4	Length checksum	–
6	PTP other	–	1	PTP version	2	1	Msg type	< 4
8	Correction	–	6	PTP other	–	1	PTP version	2
14	PTP other	–	8	Correction	–	6	PTP other	–
2	Sequence number	–	14	PTP other	–	8	Correction	–
2	PTP other	–	2	Sequence number	–	14	PTP other	–
8	TS insertion[a]	–	2	PTP other	–	2	Sequence number	–
10	Origin timestamp	–	8	TS insertion[a]	–	2	PTP other	–
–	–	–	10	Origin timestamp	–	8	TS insertion[a]	–
–	–	–	–	–	–	10	Origin timestamp	–

[a] Timestamp insertion can only be used in RX slave clock and specific grand master modes.

Variable and Fixed Latency Support

The BCM54285 can support either variable or fixed latency modes.

Y.1731 Support

The BCM54285 can recognize and timestamp Y.1731 protocol data units (PDUs) related to delay measurements in both directions for both (one-way and two-way).

Synchronous Ethernet

The BCM54285 has the provision to output a clock, recovered from the copper or fiber MDI link partner, to allow point-to-point synchronization of the clock frequency. The recovered clock output, if it is fed back into the BCM54285 REFCLKP/REFCLKN input, must be fed into a jitter attenuating cleanup PLL to attenuate accumulated jitter and provide a clean reference.

The recovered clock is 25 MHz when the BCM54285 is linked in 1000BASE-T Slave and 100BASE-TX mode, as well as 1000BASE-X and 100BASE-FX mode. When in 1000BASE-T mode, to output the recovered clock from the link partner, the BCM54285 must auto-negotiate to Slave mode. If the BCM54285 auto-negotiates to Master mode, the BCM54285 would recover its own clock, not the clock of the link partner. 10BASE-T is Manchester encoded and the clock phase information is only transmitted when a packet is being transmitted, so 10BASE-T cannot not be used for Synchronous Ethernet applications.

The BCM54285 provides two recovered clocks (REC_CLK[1] and REC_CLK[2]) and two PLL lock indicators (LOC_REC_CLK[1] and LOC_REC_CLK[2]) for Synchronous Ethernet applications. The recovered clocks will be based on either the BCM54285's reference clock or the recovered clock from one of the BCM54285's ports. Table 7 on page 33 shows the source the recovered clocks are derived from. The recovered clock output must be fed into a jitter attenuating cleanup PLL to attenuate accumulated jitter and provide a clean reference. The recovered clock is 25 MHz when the device is linked in 1000BASE-T Slave and 100BASE-TX modes or 1000BASE-X and 100BASE-FX modes. For additional information, see *Synchronizing Broadcom PHYs for Point-to-Point Synchronous Ethernet Applications* (SYNCE-AN1XX-R).

Caution! When Synchronous Ethernet is enabled, EEE and AutogrEEEn mode must be disabled on the port that is used for the REC_CLK[1] and or REC_CLK[2].

Caution! 1000BASE-T master mode recovers its own clock and should not be used for Synchronous Ethernet applications.

Caution! 10BASE-T is Manchester encoded and the clock phase information is only transmitted when a packet is being transmitted, therefore 10BASE-T should not be used for Synchronous Ethernet applications.

Table 7: Recovered Clock Source and Speed Based on Link Status

Mode	Recovered Clock Source	Recovered Clock Speed
1000BASE-T (slave, linkup)	Recovered clock from line-side data.	25 MHz
1000BASE-T (master, linkup)	BCM54285 reference clock.	25 MHz
1000BASE-T (master or slave, link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b0.	No clock.	25 MHz
1000BASE-T (master or slave, link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b1.	Driven low.	NA
100BASE-TX (linkup)	Recovered clock from line-side data.	25 MHz
100BASE-TX (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b0.	BCM54285 reference clock.	25 MHz
100BASE-TX (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b1.	Driven low.	NA
10BASE-T (linkup)	Recovered clock from line-side data.	5 MHz
10BASE-T (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b0.	BCM54285 reference clock.	25 MHz
10BASE-T (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b1.	Driven low.	NA
1000BASE-X (linkup)	Recovered clock from line-side data.	25 MHz
1000BASE-X (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b0.	BCM54285 reference clock.	25 MHz
1000BASE-X (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b1.	Driven low.	NA
100BASE-FX (linkup)	Recovered clock from line-side data.	25 MHz
100BASE-FX (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b0.	BCM54285 reference clock.	25 MHz
100BASE-FX (link-down) • RDB_Register, offset 0x83C, bit[8] = 1’b1.	Driven low.	NA

REC_CLK[1] and REC_CLK[2]

REC_CLK[1] and REC_CLK[2] are disabled by default. The port that the recovered clocks are derived from can be selected by writes to the SYNCE_RECOVERY_CLOCK register. The port that REC_CLK[1] is derived from is set by bits[2:0] and the port that REC_CLK[2] is derived from is set by bits[6:4] See Table 8. The BCM54285 also supports Sync_E auto-clock disable mode and Sync_E auto switching mode.

Table 8: Recovered Clock Port Selection

REC_CLK[1]		***REC_CLK[2]***	
Port #	***RDB_Register, offset 0x83C, Bits[2:0]***	***Port #***	***RDB_Register, offset 0x83C, Bits[6:4]***
Port 0	0x0	Port 0	0x0
Port 1	0x1	Port 1	0x1
Port 2	0x2	Port 2	0x2
Port 3	0x3	Port 3	0x3
Port 4	0x4	Port 4	0x4
Port 5	0x5	Port 5	0x5
Port 6	0x6	Port 6	0x6
Port 7	0x7	Port 7	0x7

By default, the REC_CLK[1] and REF_CLK[2] outputs are disabled.

- Enable REC_CLK[1]: REC_CLK1_DISABLE bit (RDB_Register, offset 0x83C, bit[3] = 1’b0).
- Disable REC_CLK[1]: REC_CLK1_DISABLE bit (RDB_Register, offset 0x83C, bit[3] = 1’b1).
- Enable REC_CLK[2]: REC_CLK2_DISABLE bit (RDB_Register, offset 0x83C, bit[7] = 1’b0).
- Disable REC_CLK[2]: REC_CLK2_DISABLE bit (RDB_Register, offset 0x83C, bit[7] = 1’b1).

Sync_ E Auto-Clock Disable Mode

Sync_E auto-clock disable mode programs the REC_CLK[1] and REC_CLK[2] outputs to drive low when the link is down. By default, the REC_CLK[1] and REC_CLK[2] output a 25 MHz clock based on the REFCLKP/REFCLKN clock input when the link is down. Once link is lost when in auto-clock switching mode, it will take approximately 200 ns for the REC_CLK[1] and REC_CLK[2] outputs to be driven low.

- Enable auto-clock switching mode: SYNCE_AUTO_CLK_DIS bit (RDB_Register, offset 0x83C, bit[8] = 1’b1).
- Disable auto-clock switching mode: SYNCE_AUTO_CLK_DIS bit (RDB_Register, offset 0x83C, bit[8] = 1’b0).

Sync_E Auto-Switch Mode

Enabling Sync_E auto-switch mode programs the REC_CLK[1] output to switch from the primary port’s recovered clock to the secondary port’s recovered clock when link is lost on the primary port. REF_CLK[2] output will still be based on the secondary port’s recovered clock.

- The primary port is determined by REC_CLK1_SEL bits (RDB_Register, offset 0x83C, bits[2:0]).
- The secondary port is determined by REC_CLK2_SEL bits (RDB_Register, offset 0x83C, bits[6:4]).

The switch-over from the primary port’s recovered clock to the secondary port’s recovered clock takes approximately 240 ns after the link goes down on the primary port. By default, Sync_E auto-switching mode is disabled.

- Enable Sync_E auto-switch mode: SYNCE_AUTO_SW_MODE bit (RDB_Register, offset 0x83C, bit[9] = 1’b1.)
- Disable Sync_E auto-switch mode: SYNCE_AUTO_SW_MODE bit (RDB_Register, offset 0x83C, bit[9] = 1’b0.)

LOC_REC_CLK[1] and LOC_REC_CLK[2]

When in Sync_E auto-clock disable mode:

- LOC_REC_CLK[1] signal will go high when REC_CLK[1] is locked to the primary port’s recovered clock.
- LOC_REC_CLK[1] signal will go low when REC_CLK[1] is not locked to the primary port’s recovered clock or the link is lost.
- LOC_REC_CLK[2] signal will go high when REC_CLK[2] is locked to the secondary port’s recovered clock.
- LOC_REC_CLK[2] signal will go low when REC_CLK[1] is not locked to the secondary port’s recovered clock or the link is lost.

When in Sync_E auto-switching mode:

- LOC_REC_CLK[1] signal will go high when REC_CLK[1] is locked to the primary port’s recovered clock, or when link is lost on the primary port and link is still up on the secondary port. LOCK_REC_CLK[1] will reflect the status of the secondary port’s recovered clock when link is down on the primary port.
- LOC_REC_CLK[1] signal will go low when REC_CLK[1] primary port’s is not locked to recovered clock, or the link is lost and the REC_CLK[2] secondary port is not locked to the recovered clock or link is lost.
- LOC_REC_CLK[2] signal will go high when REC_CLK[2] is locked to the secondary port’s recovered clock.
- LOC_REC_CLK[2] signal will go low when REC_CLK[1] is not locked to the secondary port’s recovered clock or the link is lost.

Start of Frame

When receiving packets from the line-side (copper/fiber interface), the RX_SOP signal will go high at the start-of-frame- deliminator (SFD) and remain active high until the end-of-stream-deliminator (ESD).

When receiving packets from the system-side (QSGMII interface) the TX_SOP signal will go high at the start-of-frame-deliminator (SFD) and remain active high until the end-of-stream-deliminator.

Receive start-of-packet signal RX_SOP is available on LED[2] by programming RDB_Register, offset 0x01E, Bits[7:4] = 0x9. Transmit start-of-packet signal TX_SOP is available on LED[3] by programming RDB_Register, offset 0x01E, Bits[3:0] = 0x9.

Cleanup PLL Considerations

The input clock jitter to the BCM543XXCT must be:

- For input clocks = 25 MHz the input clock jitter must be < 1.5ps-rms @ Fj = 1 kHz to 5 MHz offset and the clock PPM must be < +50 ppm.

- For input clocks > 25 MHz the input clock jitter must be < 1.5ps-rms @ Fj = 12 kHz to 20 MHz offset and the clock PPM must be < +50 ppm.

Jitter attenuators which meet this requirement include Valpey Fisher® VFJA910 and Pericom® PI6CX201A.

Fast Link Drop Detection

Some Synchronous Ethernet applications may require detection and recovery from link impairments quickly. Per IEEE 802.3 Clause 40, the link drop times are:

- 1000BASE-T Master: 750 ms
- 1000BASE-T Slave: 350 ms
- 100BASE-TX: 60 µs
- 10BASE-T: 100 ms

The 1000BASE-T link drop detect times can be decreased by using the LOCAL_RCVR_STAT_CHANGE bit in the COPPER_INTERRUPT_MASK register (RDB_Register, offset 0x00B, bit[4]) to determine a Link Fail condition within 1 ms. Setting bit[4] = 1'b0 will cause any LED ball programmed for INTR to go high when the link goes down, within 1 ms. Multiple LED balls programmed for INTR can be connected together.

Note: Using this method does not change the time it takes for the link to go down. It only provides a method of detecting a possible link-down scenario that is faster than using the LINK-Down indicators in the PHY.
EEE and AutogrEEEn must be disabled on the primary and/or secondary ports.

Section 4: Interfaces

Overview

This section describes the various interfaces that the BCM54285 supports.

- "QSGMII" on page 37
- "Copper Interface" on page 38
- "Fiber Interface" on page 49
- "Management Interface" on page 54
- "Interrupt Interface" on page 59
- "LED Interface" on page 62
- "Broadcom Serial Control Interface" on page 69

QSGMII

The BCM54285 can communicate with Ethernet switches that support a QSGMII. The QSGMII transmits serial data differentially at 5 Gbaud via TDP0, TDN0, TDP1 and TDN1, and receives serial data differentially via RDP0, RDN0, RDP1, and RDN0. Each differential pair has an 80Ω to 120Ω differential impedance. The BCM54285 recovers the clock from the QSGMII input data. The BCM54285 has two QSGMII cores. Each core supports four SGMII lanes. When configured to QSGMII mode, the BCM54285 copper interface auto-negotiates based on Clauses 28 and 40 of IEEE 802.3. The QSGMII auto-negotiates according to 1000BASE-X auto-negotiation as described in IEEE 802.3, Clause 37, except for a few changes to operate in QSGMII mode. The link timer is reduced to 1.6 ms, and the auto-negotiation code word is changed to reflect the copper link, copper duplex, and copper speed. QSGMII, based on the PHY port's MDI negotiation, auto-negotiates per clause 37 in IEEE 802.3 with several modifications for QSGMII operation.

Table 9: QSGMII Interface Balls

QSGMII Signal Ball	*Description*
• RDP0 and RDN0 • RDP1 and RDN0	• QSGMII differential input for ports 0 to 3. • QSGMII differential input for ports 4 to 7. Differential 5.0 Gbaud data from the switch to the BCM542XXCT. These input balls have an on-chip internal 80Ω to 120Ω differential impedance. It is highly recommended to use 0.01 µF to 0.1 µF coupling capacitors be used.
• TDP0 and TDN0 • TDP1 and TDN1	• QSGMII differential output for ports 0 to 3. • QSGMII differential output for ports 4 to 7. Differential 5.0 Gbaud output data from the BCM542XXCT to the switch.

Copper Interface

The BCM54285 can communicate with link partners that support 10BASE-T, 100BASE-TX or 1000BASE-T. The BCM54285 supports auto-negotiation for 10BASE-T, 100BASE-TX or 1000BASE-T. The BCM54285 supports force mode for 10BASE-T and 100BASE-TX. Force mode is not supported for 1000BASE-T operation.

The following sections describe the internal circuitry and additional features of the copper interface.

Encoder

In 10BASE-T mode, Manchester encoding is performed on the data stream that is transmitted on the twisted-pair cable. The multimode transmit digital-to-analog converter (DAC) performs preequalization for 100m of Category 3 cabling.

In 100BASE-TX mode, the BCM54285 transmits a continuous data stream over the twisted-pair cable. The transmit packet is encapsulated by replacing the first 2 nibbles of preamble with a start-of-stream delimiter (/J/K codes) and appending an end-of-stream delimiter (/T/R codes) to the end of the packet. The transmitter repeatedly sends the idle code group between packets. The encoded data stream is serialized and then scrambled by the stream cipher block, as described in "Stream Cipher" on page 41. The scrambled data is then encoded into MLT-3 signal levels.

In 1000BASE-T mode, the BCM54285 simultaneously transmits and receives a continuous data stream on all four pairs of the Category 5 cable. Byte-wide data from the transmit data signals are scrambled when the transmit enable is asserted, and the trellis (a PAM5 symbol on each of the four twisted-pairs) is encoded into a four-dimensional code group and then inserted into the transmit data stream. The transmit packet is encapsulated by replacing the first 2-bytes of the preamble with a start-of-stream delimiter, and appending an end-of-stream delimiter to the end of the packet. When the transmit error input is asserted during a packet transmission, a transmit error code group is sent in place of the corresponding data code group. The transmitter sends idle code groups or carrier extend code groups between packets. Carrier extension is used by the switch to separate packets within a multiple-packet burst, and is indicated by asserting the transmit error signal and placing 0Fh on the transmit data signals while the transmit enable is low. A carrier extend error is indicated by replacing the transmit data input with 0x1F during carrier extension. The encoding complies with IEEE 802.3ab and is fully compatible with previous versions of the Broadcom 1000BASE-T PHYs.

Decoder

In 10BASE-T mode, Manchester decoding is performed on the data stream.

In 100BASE-TX mode, following equalization and clock recovery, the receive data stream is converted from MLT-3 to serial nonreturn to zero (NRZ) data. The NRZ data is descrambled by the stream cipher block, as described later in this document. The descrambled data is then deserialized and aligned into 5-bit code groups. The 5-bit code groups are decoded into 4-bit data nibbles. The start-of-stream delimiter is replaced with preamble nibbles, and the end-of-stream delimiter and idle codes are replaced with zeros. The decoded data is driven onto the MII receive data outputs. When an invalid code group is detected in the data stream, the BCM54285 asserts the MII receive error (RX_ER) signal. RX_ER is also asserted when the link fails or when the descrambler loses lock during packet reception.

In 1000BASE-T mode, the receive data stream is:

- Passed through the Viterbi decoder
- Descrambled
- Translated back into byte-wide data

The start-of-stream delimiter is replaced with preamble bytes, and the end-of-stream delimiter and idle codes are replaced with 0x00. Carrier extend codes are replaced with 0x0F or 0x1F. The decoded data is driven onto the QSGMII receive data outputs. Decoding complies with IEEE 802.3ab and is fully compatible with previous versions of Broadcom 1000BASE-T PHYs.

Link Monitor

In 10BASE-T mode, a link-pulse detection circuit constantly monitors the following pairs for the presence of valid link pulses.

- TDP_[7:0]_[0]/TDN_[7:0]_[0]
- TDP_[7:0]_[1]/TDN_[7:0]_[1]

In 100BASE-TX mode, receive signal energy is detected by monitoring the receive pair for transitions in the signal level. Signal levels are qualified using squelch detect circuits. When no signal is detected on the receive pair, the link monitor enters the Link Fail state, and the transmission and reception of data packets are disabled. When a valid signal is detected on the receive pair for a minimum of 1 ms, the link monitor enters the Link Pass state, and the transmit and receive functions are enabled.

Following auto-negotiation in 1000BASE-T mode, the master transceiver begins sending data on the media. The slave transceiver also begins transmitting when it has recovered the master transceiver’s timing. Each end of the link continuously monitors its local receiver status. When the local receiver status has been good for at least 1 μs, the link monitor enters the Link Pass state, and the transmission and reception of data packets are enabled. When the local receiver status is bad for more than 750 ms, the link monitor enters the Link Fail state, and the transmission and reception of data packets are then disabled.

Digital Adaptive Equalizer

The digital adaptive equalizer removes intersymbol interference (ISI) created by the transmission channel media. The equalizer accepts sampled unequalized data from the analog-to-digital converter (ADC) on each channel and produces equalized data. The BCM54285 achieves an optimum signal-to-noise ratio by using a combination of feed forward equalization (FFE) and decision feedback equalization (DFE) techniques. Under harsh noise environments, these powerful techniques achieve a bit error rate (BER) of less than 1×10^{-12} for transmissions of: up to 100 meters on Category 5 twisted-pair cabling for 1000BASE-T and 100BASE-TX mode; up to 100 meters on Category 3 UTP cable for 10BASE-T mode. The all-digital nature of the design makes the BCM54285 very tolerant to noise. The filter coefficients are self-adapting to accommodate varying conditions of cable quality and cable length.

Echo Canceler

Because of the bidirectional nature of the channel in 1000BASE-T, an echo impairment is caused by each transmitter. The output of the echo filter is added to the FFE output to remove the transmitted signal impairment from the incoming receive signal. The echo canceler coefficients are self-adapting to manage the varying echo impulse responses caused by different channels, transmitters, and environmental conditions.

Crosstalk Canceler

The BCM54285 transmits and receives a continuous data stream on four channels in gigabit mode. For a given channel, the signals sent by the other three local transmitters cause impairments on the received signal because of near-end crosstalk (NEXT) between the pairs. It is possible to cancel the effect because each receiver has access to the data for the other three pairs that cause this interference. The output of the adaptive NEXT canceling filters is added to the FFE output to cancel the NEXT impairment.

Analog-to-Digital Converter

Each receive channel has its own 125 MHz analog-to-digital converter (ADC) that samples the incoming data on the receive channel and feeds the output to the digital adaptive equalizer. Advanced analog circuit techniques achieve the following results:

- Low offset
- High-power supply noise rejection
- Fast settling time
- Low bit error rate

Clock Recovery/Generator

The clock recovery and generator block creates the transmit and receive clocks for 1000BASE-T, 100BASE-TX, and 10BASE-T operation.

In 10BASE-T or 100BASE-TX mode, the transmit clock is locked to the reference clock input, and the receive clock is locked to the incoming data stream.

In 1000BASE-T mode, the two ends of the link perform loop timing. One end of the link is configured as the master, and the other is configured as the slave. The master transmit and receive clocks are locked to the reference clock input. The slave transmit and receive clocks are locked to the incoming receive data stream. Loop timing allows for the cancellation of echo and NEXT impairments by ensuring that the transmitter and receiver at each end of the link are operating at the same frequency.

Baseline Wander Correction

1000BASE-T and 100BASE-TX data streams are not always DC-balanced. Because the receive signal must pass through a transformer, the DC offset of the differential receive input can vary with data content. This effect, which is known as baseline wander, can greatly reduce the noise immunity of the receiver. The BCM54285 automatically compensates for baseline wander by removing the DC offset from the input signal, significantly reducing the probability of a receive symbol error.

In 10BASE-T mode, baseline wander correction is not performed because the Manchester coding provides a perfect DC balance.

Multimode TX Digital-to-Analog Converter

The multimode transmit digital-to-analog converter (DAC) transmits PAM5, MLT-3, and Manchester coded symbols. The transmit DAC performs signal wave shaping that decreases the unwanted high frequency signal components, reducing electromagnetic interference (EMI). The transmit DAC uses a voltage drive output that is well-balanced and produces very low-noise transmit signals.

Stream Cipher

In 1000BASE-T and 100BASE-TX modes, the transmit data stream is scrambled to reduce radiated emissions and to ensure that there are adequate transitions within the data stream. The 1000BASE-T scrambler also ensures that there is no correlation among symbols on the four different wire pairs and in the transmit and receive data streams. The scrambler reduces peak emissions by randomly spreading the signal energy over the transmit frequency range and eliminating peaks at certain frequencies. The randomization of the data stream also assists the digital adaptive equalizers and echo/crosstalk cancelers. The algorithms in these circuits require that there be no sequential or cross-channel correlation among symbols in the various data streams.

In 100BASE-TX mode, the transmit data stream is scrambled by exclusive ORing the encoded serial data stream. This is done with the output of an 11-bit wide linear feedback shift register (LFSR), producing a 2047-bit nonrepeating sequence.

In 1000BASE-T mode, the transmit data stream is scrambled by exclusive ORing the input data byte with an 8-bit wide cipher text word. The cipher text word generates each symbol period from 8 uncorrelated maximal length data sequences that are produced by linear remapping of the output of a 33-bit wide LFSR. After the scrambled data bytes are encoded, the sign of each transmitted symbol is again randomized by a 4-bit wide cipher text word that is generated in the same manner as the 8-bit word. The master and slave transmitters use different scrambler sequences to generate the cipher text words. For repeater or switch applications, where all ports can transmit the same data simultaneously, signal energy is randomized further by using a unique seed to initialize the scrambler sequence for each PHY.

The receiver descrambles the incoming data stream by exclusive ORing it with the same sequence generated at the transmitter. The descrambler detects the state of the transmit LFSR by looking for a sequence representing consecutive idle code groups. The descrambler locks to the scrambler state after detecting a sufficient number of consecutive idle codes. The BCM54285 enables transmission and reception of packet data only when the descrambler is locked. The receiver continually monitors the input data stream to ensure that it has not lost synchronization by checking that inter-packet gaps containing idles or frame extensions are received at expected intervals. When the BCM54285 detects loss of synchronization, it notifies the link partner of the inability to receive packets (1000BASE-T mode only) and attempts to resynchronize to the received data stream. If the descrambler is unable to resynchronize for a period of 750 ms, the BCM54285 is forced into the Link Fail state.

In 10BASE-T mode, scrambling is not required to reduce radiated emissions.

Wiremap and Pair Skew Correction

During 1000BASE-T operation, the BCM54285 can automatically detect and correct some UTP cable wiring errors. The symbol decoder detects and compensates for the following errors at the PHY’s receiver:

- Wiring errors caused by the swapping of pairs within the UTP cable. It can correct for (Pair A and B swap) and (Pair C and D swap). It cannot compensate for (Pair A and D swap), (Pair A and C swap), (Pair B and C swap) or (Pair B and D swap).
- Polarity errors caused by the swapping of wires within a pair.
- Delay skews caused by UTP cable pair propagation delay differences or PCB trace length differences. The BCM54285 automatically compensates for differences in the arrival times of symbols on the four pairs of the UTP cable. The varying arrival times are caused by differing propagation delays (commonly referred to as delay skew) between the cable pairs. The BCM54285 can tolerate delay skews of up to 64 ns long.

During 10/100 Mbps operation, the BCM54285 can automatically detect and correct some UTP cable wiring errors. The symbol decoder detects and compensates for the following errors at the PHY’s receiver:

- Wiring errors caused by the swapping of pairs within the UTP cable. It can correct for (Pair A and B swap).
- Polarity errors caused by the swapping of wires within a pair.

Delay skew is not an issue for 10BASE-T or 100BASE-TX, since only one pair is used in each direction.

⚠ **Caution!** The Twisted Pair interface connection to the RJ45 should always be done per the IEEE 802.3 specification even though the PHY can compensate for most errors in wiring. The reason is no compensation is done on the PHY’s transmitter. If a Link Partner can’t compensate for pair swaps or polarity errors, then the Link Partner will never be able to link-up.

Automatic MDI Crossover

During copper auto-negotiation, one end of the link needs to perform an MDI crossover so that each transceiver’s transmitter is connected to the other receiver. The BCM54285 can perform an automatic media dependent interface (MDI) crossover, eliminating the need for crossover cables or cross-wired (MDIX) ports. During auto-negotiation, the BCM54285 normally transmits on TDP_[7:0]_0/TDN_[7:0]_0 and receives on TDP_[7:0]_1/TDN_[7:0]_1.

When connecting to another device that does not perform MDI crossover, the BCM54285 automatically switches its TDP_[7:0]_0/TDN_[7:0]_0 and TDP_[7:0]_1/TDN_[7:0]_1 pairs when necessary to communicate with the remote device. When connecting to another device that does have MDI crossover capability, an algorithm determines which end performs the crossover function.

The MDI Crossover State can be determined by reading RDB_Register, offset 0x001, bit[13].

- 1’b0 = Normal MDI mode
- 1’b1 = Crossover MDI mode

1000BASE-T Operation

During 1000BASE-T operation, the BCM54285 swaps the transmit symbols on pairs 0 and 1 and pairs 2 and 3 if auto-negotiation completes in the MDI crossover state. The 1000BASE-T receiver automatically detects pair swaps on the receive inputs and aligns the symbols properly within the decoder. The automatic MDI crossover function cannot be disabled when in 1000BASE-T mode.

10/100BASE-TX Operation (Auto-Negotiation Enabled)

During 10BASE-T and 100BASE-TX operation, pair swaps automatically occur within the device and do not require user intervention. The automatic MDI crossover function, by default, only works when auto-negotiation is enabled. This function can be disabled during auto-negotiation by writing to RDB_Register, offset 0x000, bit[14] = 1’b1.

10/100BASE-TX Operation (Forced Mode)

The automatic MDI crossover function can also be enabled when in forced 10BASE-T or forced 100BASE-TX mode. This feature allows the user to disable the copper auto-negotiation in either 10BASE-T or 100BASE-TX and still take advantage of the automatic MDI crossover function. This feature is enabled by writing RDB_Register, offset 0x02F, bit[9] = 1’b1.

When in forced 10BASE-T or 100BASE-TX mode, the BCM54285 has a feature that can manually swap the MDI state when the automatic MDI crossover function is disabled. Normally the BCM54285 transmits on TDP_[7:0]_[0]/TDN_[7:0]_[0] and receives on TDP_[7:0]_[1]/TDN_[7:0]_[1]. To change the MDI state to transmit on TDP_[7:0]_[1]/TDN_[7:0]_[1] and receive on TDP_[7:0]_[0]/TDN_[7:0]_[0] the following steps must be done.

- Put PHY in nonlink condition.
- Enable Manual Swap MDI (Write RDB_Register, offset 0x00E, bit[7] = 1’b1).
- Set PHY into Force 10BASE-T or 100BASE-TX mode.

Note: To change the MDI state when in forced 100BASE-TX mode, the PHY must first be put into a nonlink condition.

Full-Duplex Mode

The BCM54285 supports full-duplex operation. While in full-duplex mode, a transceiver can simultaneously transmit and receive packets on the cable. When auto-negotiation is disabled, full-duplex operation can be enabled by setting register 0x00, bit[8] = 1’b1.

When auto-negotiation is enabled, full-duplex capability is advertised for:

- 10BASE-T: Register 0x04, bit[6] = 1’b1.
- 100BASE-TX: Register 0x04, bit[8] = 1’b1.
- 1000BASE-T: Register 0x09, bit[9] = 1’b1.

Master/Slave Configuration

In 1000BASE-T mode, the BCM54285 and its link partner perform loop timing. One end of the link must be configured as the timing master and the other end as the slave. Master/slave configuration is performed by the auto-negotiation function. The auto-negotiation function first looks at the manual master/slave configuration bits advertised by the local PHY and the link partner. If neither PHY requests manual configuration, then the auto-negotiation function looks at the advertised repeater/DTE settings. If one PHY is advertised as a repeater port and the other is advertised as a DTE port, then the repeater port is configured as the master and the DTE port the slave. Each end generates an 11-bit random seed if the two settings are equal; the end with the higher seed is configured as the master. If the local PHY and the link partner generate the same random seed, then auto-negotiation is restarted.

If both ends of the link attempt to force the same manual configuration (both master or both slave), or the random seeds match seven consecutive times, then the BCM54285 sets the Master/Slave Configuration Fault bit in the 1000BASE-T Status register and auto-negotiation is restarted.

For setting the BCM54285 to manual master/slave configuration or to set the advertised repeater/DTE configuration, see 1000BASE-T control register (Address 0x09).

Next Page Exchange

The 1000BASE-T configuration requires the exchange of three auto-negotiation next pages between the BCM54285 and its link partner. Exchange of 1000BASE-T Next Page information takes place automatically when the BCM54285 is configured to advertise 1000BASE-T capability.

The BCM54285 also supports software-controlled Next Page exchanges. When register 0x04, bit[15] = 1’b1, all Next Page transactions are controlled through the MII management interface. This includes the three 1000BASE-T Next Pages, which are always sent first. The BCM54285 automatically generates the appropriate message code field for the 1000BASE-T pages. When the BCM54285 is not configured to advertise 1000BASE-T capability, the 1000BASE-T Next Pages are not sent.

When the BCM54285 is not configured to advertise 1000BASE-T capability and register 0x04, bit[15] = 1’b0, the BCM54285 does not advertise Next Page ability.

Auto-Negotiation

The BCM54285, when configured to copper mode, negotiates its mode of operation over the copper media using the auto-negotiation mechanism, defined in the IEEE 802.3u and 802.3ab specifications. When the auto-negotiation function is enabled, the BCM54285 automatically chooses the mode of operation by advertising its abilities and comparing them with those received from its link partner. The BCM54285 can be configured to advertise the following modes:

- 1000BASE-T full-duplex and/or half-duplex.
- 100BASE-TX full-duplex and/or half-duplex.
- 10BASE-T full-duplex and/or half-duplex.

The transceiver negotiates with its link partner and chooses the highest common operating speed and duplex mode, commonly referred to as highest common denominator (HCD). Auto-negotiation can be enabled or disabled by hardware and software control, but is always required for 1000BASE-T operation.

Ethernet@Wirespeed

Ethernet@Wirespeed is an enhancement to auto-negotiation that allows a network connection over impaired cable plants. If a link cannot be established at the highest common denominator within a set number of link attempts, then the BCM54285 advertises the next highest advertised speed using auto-negotiation. The set number of failed link attempts is programmable. See “Changing the Number of Failed Link Attempts” on page 46 for details.

The BCM54285 has a link-up timer that times how long the link has been up. If the link stays up for less than 3 seconds, then the Link-Fail Counter will get incremented. If the link stays up for greater than 5 seconds, then the Link-Fail Counter is reset to zero.

The purpose of the linkup timer is to prevent scenarios where an unstable link (link is going up and down quickly) causes the BCM54285 to continuously try to link at a given speed and not try to downgrade and link to a lower speed. In this situation, if the link is up for less than 3 seconds, the Link-Fail Counter is incremented. Once the Link-Fail Counter exceeds the programmable failed link attempts the BCM54285 will start advertising the next lowest speed and try to establish a link.

The linkup timer can be bypassed by setting RDB_Register, offset 0x02F, bit[10] = 1’b1. Setting this bit causes the number of failed link attempts to get reset to zero after every link up condition, no matter how short the linkup time is.

Ethernet@Wirespeed Example

At start-up the BCM54285 is advertising 1000BASE-T, 100BASE-TX, and 10BASE-T capabilities per register 0x04 and register 0x09, and the Link Partner is also advertising the same capabilities:

- If a link cannot be established within a programmable number of link attempts (two to nine) with 1000BASE-T being advertised, then an Ethernet@Wirespeed downgrade occurs, the 1000BASE-T capability is masked out on the BCM54285 and the next highest advertised capability (100BASE-TX) is advertised.

- If a link cannot be established within a programmable number of link attempts (two to nine) with 100BASE-TX being advertised, then an Ethernet@Wirespeed downgrade occurs, the 100BASE-TX is masked out on the BCM54285 and the next highest advertised capability (10BASE-T) is advertised.
- If a link cannot be established within a programmable number of link attempts (two to nine) with 10BASE-T being advertised, then an Ethernet@Wirespeed downgrade occurs and all advertising capabilities are enabled (1000BASE-T, 100BASE-TX, and 10BASE-T) on the BCM54285 and the whole process begins again.

Enabling/Disabling Ethernet@Wirespeed

Enabling or disabling Ethernet@Wirespeed is done on a per port basis.

- Enable: Write RDB_Register, offset 0x02F, bit[4] = 1’b1.
- Disable: Write RDB_Register, offset 0x02F, bit[4] = 1’b0.

Removing Ethernet@Wirespeed Downgrade

Ethernet@Wirespeed downgrade can be removed by any of the following events:

- Stable linkup condition for greater than 5 seconds.
- Unplug cable (no energy) for 6 seconds.
- Hardware reset.
- Software reset (write register 0x00, bit[15] = 1’b1).
- Disable auto-negotiation (write register 0x00, bit[12] = 1’b0).
- Restart auto-negotiation (Write register 0x00, bit[9] = 1’b1).
- Disabling Wirespeed (Write RDB_Register, offset 0x02F, bit[4] = 1’b0).
- Auto-negotiation resolves to no highest common denominator (HCD).

Changing the Number of Failed Link Attempts

The number of failed link attempts before downgrading to a slower speed is programmable. The number can be programmed anywhere from two to nine failed link attempts before downgrading to a lower speed. The default value is five failed link attempts. The number of failed link attempts before downgrading to a lower speed can be programmed by writing to RDB_Register, offset 0x014, bits[4:2] as shown Table 10 on page 46.

Table 10: Changing the Number of Failed Link Attempts before Downgrade

Bits[4:2]	***Description***
0x0	Number of failed link attempts before Ethernet@Wirespeed downgrade = 2
0x1	Number of failed link attempts before Ethernet@Wirespeed downgrade = 3
0x2	Number of failed link attempts before Ethernet@Wirespeed downgrade = 4
0x3	Number of failed link attempts before Ethernet@Wirespeed downgrade = 5 (default value)
0x4	Number of failed link attempts before Ethernet@Wirespeed downgrade = 6
0x5	Number of failed link attempts before Ethernet@Wirespeed downgrade = 7
0x6	Number of failed link attempts before Ethernet@Wirespeed downgrade = 8
0x7	Number of failed link attempts before Ethernet@Wirespeed downgrade = 9

Monitoring Ethernet@Wirespeed

The status of the Ethernet@ Wirespeed downgrade can be monitored through the following registers and LEDs.

- Ethernet@Wirespeed Downgrade Status (Read RDB_Register, offset 0x001, bit[14]).
- Ethernet@Wirespeed Downgrade (Read RDB_Register, offset 0x00C, bit[12]).
- Ethernet@Wirespeed Disable Gigabit Advertising (Read RDB_Register, offset 0x00C, bit[14]).
- Ethernet@Wirespeed Disable 100BASE-TX Advertising (Read RDB_Register, offset 0x00C, bit[13]).
- HCD Status (Read RDB_Register, offset 0x00C, bits[11:0]).
- Auto-negotiation HCD and Current Status (Read RDB_Register, offset 0x009, bits[10:8]).
- Ethernet@Wirespeed downgrade LED on LED[0] (Write RDB_Register, offset 0x01D, bits[3:0] = 0x9).
- Ethernet@Wirespeed downgrade LED on LED[1] (Write RDB_Register, offset 0x01D, bits[7:4] = 0x9).

Super Isolate Mode

When in Super Isolate mode, the following happens:

- The BCM54285's transmitter and receiver on the copper media dependent Interface are disabled. The link partner will go into a link down state since it is not receiving any FLPs, NLPs, or 100BASE-TX idles.
- The BCM54285's QSGMII interface will be in QSGMII auto-negotiation mode. A QSGMII link between the BCM54285 and the switch is established if the switch sends back an acknowledgement to the BCM54285 through the QSGMII auto-negotiation link code word. If the switch does not send back an acknowledgement, the PHY remains in QSGMII auto-negotiation mode.

Hardware Enable

The BCM54285 has an additional hardware-strapping ball that can put the BCM54285 into Super Isolate mode right after a software or hardware reset. When Super Isolate mode is entered via the hardware-strapping ball, all 8 ports are affected. To enable Super Isolate mode, pull SUPER_I high at reset. The Super Isolate ball is sampled at reset and has an internal pull-down resistor. Super Isolate mode can only be disabled by software on a per port basis.

Software Enable/Disable

The BCM54285 can be put into Super Isolate mode on a per port basis by software.

- To enable Super Isolate mode:
  Write RDB_Register, offset 0x02A, bit[5] = 1’b1 for each of the 8 ports.
- To disable Super Isolate mode:
  Write RDB_Register, offset 0x02A, bit[5] = 1’b0 for each of the 8 ports.

Standby Power-Down Mode

The BCM54285 can be placed into standby Power-down mode using software commands. In this mode, all PHY functions, except for the serial management interface, are disabled. To enter standby Power-down mode, write register 0x00, bit[11] = 1’b1. There are three ways to exit standby Power-down mode:

- Write register 0x00, bit[11] = 1’b0 (Clear MII Control register)
- Write register 0x00, bit[15] = 1’b1 (Software reset)
- Assert the hardware $\overline{\text{RESET}}$

Reads or writes to any MII register, other than register 0x00 while the device is in the standby Power-down mode, returns unpredictable results. Upon exiting standby Standby Power-down mode, the BCM54285 remains in an internal reset state for 40 µs, and then resumes normal operation.

Auto Power-Down Mode

When the BCM54285 is placed into Auto Power-Down (APD) mode the chip power is reduced when the signal from the copper link partner is not present. APD mode works whether the device is in auto-negotiation enabled or in forced mode. When APD mode is enabled, the BCM54285 automatically enters the low-power mode when energy on the line is lost, and it resumes normal operation when energy is detected. When the BCM54285 is in APD mode, the copper transmitter is disabled (Sleep Cycle) for 2.7 seconds or 5.4 seconds depending on the SLEEP_TIMER_SEL bit after which the transmitter is enabled (Wake Cycle) for a duration of 84 ms to 1260 ms depending on the settings on the WAKE_UP_TIMER_SEL bits. The BCM54285 enters normal operation and establishes a link if energy is detected, otherwise, the Sleep and Wake-up cycles repeat.

ADP Mode Enable (Auto-Negotiation Enabled)

- Write RDB_Register, offset 0x01A, bits[6:5] = 2’b01.
- Write RDB_Register, offset 0x1A, bit[8] = 1’b’1.

ADP mode enable (auto-negotiation disabled):

- Write RDB_Register, offset 0x01A, bits[6:5] = 2’b11.
- Write RDB_Register, offset 0x1A, bit[8] = 1’b’1.

ADP mode disable:

- Write RDB_Register, offset 0x01A, bits[6:5] = 2’b00.

Sleep Cycle Settings

Write RDB_Register, offset 0x01A, bit[4]:

- 1’b0 = Disable copper transmitter for 2.7 seconds.
- 1’b0 = Disable copper transmitter for 5.4 seconds.

Wake Cycle Settings

Write register RDB_Register, offset 0x01A, bits[3:0]:

- 0x1 = Enable copper transmitter for 84 ms.

- 0x2 = Enable copper transmitter for 168 ms.
- 0x3 = Enable copper transmitter for 252 ms.
- 0xF = Enable copper transmitter for 1.26 seconds.

Fiber Interface

The BCM54285 can communicate with Link Partners that support 100BASE-FX, 1000BASE-X or SGMII-Slave. The BCM54285 supports auto-negotiation for 1000BASE-X and SGMII-Slave. The BCM54285 supports force mode for 100BASE-FX and 1000BASE-X. The following sections describe the internal circuitry and additional features of the fiber interface. Table 11 shows the fiber interface ball description.

Table 11: Fiber Interface Ball Description

Fiber Signal Pins	***Description***
SRXDN[0], SRXDP[0] SRXDN[1], SRXDP[1] SRXDN[2], SRXDP[2] SRXDN[3], SRXDP[3] SRXDN[4], SRXDP[4] SRXDN[5], SRXDP[5] SRXDN[6], SRXDP[6] SRXDN[7], SRXDP[7]	**Fiber Differential Input:** Differential 100BASE-FX or 1000BASE-X data from the Link Partner to the BCM54285. These inputs have an on-chip internal 80Ω to 120Ω differential impedance. It is highly recommended to use 0.01 µF to 0.1 µF coupling capacitors be used.
STXDN[0], STXDP[0] STXDN[1], STXDP[1] STXDN[2], STXDP[2] STXDN[3], STXDP[3] STXDN[4], STXDP[4] STXDN[5], STXDP[5] STXDN[6], STXDP[6] STXDN[7], STXDP[7]	**Fiber Differential Output:** Differential 100BASE-FX or 1000BASE-X data from the BCM54285 to the Link Partner.

100BASE-FX

When the BCM54285 is in 100BASE-FX mode it uses 4B/5B encoding and NRZI line coding. Auto-negotiation is not supported. It supports the Fiber Unidirectional mode and Far-End Fault.

100BASE-FX is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b0 (Disable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:0] = 3’b010 (Enable Fiber mode and access Copper Registers)
- Write Register 0x0, bit[11] = 1’b1 (Power down the Copper interface)
- Write RDB_Register, offset 0x233, bit[0] = 1’b1 (Enable 100BASE-FX mode)

1000BASE-X

When the BCM54285 is in 1000BASE-X mode it uses 8B/10B encoding and NRZ line coding. Auto-negotiation, force mode, and Fiber Unidirectional are supported.

1000BASE-X force mode is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b0 (Disable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:0] = 3’b010 (Enable Fiber mode and access Copper Registers)
- Write Register 0x00, bit[11] = 1’b1 (Power down the Copper interface)
- Write RDB_Register, offset 0x021, bit[0] = 1’b1 (Access Fiber Registers)
- Write Register 0x00 = 0x0140 (Force 1000BASE-X Full-Duplex mode)

1000BASE-X auto-negotiation mode is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b0 (Disable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:0] = 3’b010 (Enable Fiber mode and access Copper Registers)
- Write Register 0x00, bit[11] = 1’b1 (Power down the Copper interface)
- Write RDB_Register, offset 0x021, bit[0] = 1’b1 (Access Fiber Registers)
- Write Register 0x00 = 0x1140 (Enable Auto-Negotiation)
- Write Register 0x04: (1000BASE-X Advertisement register)
  - Bit[15] = Next Page Abilities
  - Bits[13:12] = Remote Fault Capabilities
  - Bits[8:7] = Pause Capabilities
  - Bit[6] = 1000BASE-X Half-Duplex Support
  - Bit[5] = 1000BASE-X Full-Duplex Support
- RDB_Register, offset 0x236, Bit[1] = 1’b1 (**Optional:** Enable parallel detection. Turn auto-negotiation on/off in order to link-up with link partner that is in forced 1000BASE-X mode.

SGMII-Slave

When the BCM54285 is in SGMII-Slave mode it uses 8B/10B encoding and NRZ line coding. This mode is used when connecting to a SFP SGMII to Copper Transceiver (10/100/1000BASE-T) module.

SGMII-Slave mode is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b0 (Disable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:0] = 3’b010 (Enable Fiber mode and access Copper Registers)
- Write Register 0x0, bit[11] = 1’b1 (Power down the Copper interface)
- Write RDB_Register, offset 0x235, bit[1] = 1’b1 (Enable SGMII-Slave Mode)

Auto Detect Mode

The BCM54285 can automatically switch between Copper and Fiber modes. Below are the modes that the PHY can differentiate from:

- Copper(10/100/1000BASE-T) and 100BASE-FX
- Copper(10/100/1000BASE-T) and 1000BASE-X
- Copper(10/100/1000BASE-T) and SGMII-Slave
- Copper(10/100/1000BASE-T) and 1000BASE-X/SGMII-Slave
- 1000BASE-X and SGMII-Slave

It cannot differentiate from:

- 100BASE-FX and 1000BASE-X
- 100BASE-FX and SGMII-Slave

Copper(10/100/1000BASE-T) and 100BASE-FX auto-detection is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b1 (Enable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:1] = 2’b01 (Enable Fiber mode)
- Write RDB_Register, offset 0x235, bit[1] = 1’b0 (Disable SGMII-Slave Mode)
- Write RDB_Register, offset 0x233, bit[0] = 1’b1 (Enable 100BASE-FX mode)

Copper(10/100/1000BASE-T) and 1000BASE-X auto-detection is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b1 (Enable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:1] = 2’b01 (Enable 1000BASE-X mode)
- Write RDB_Register, offset 0x233, bit[0] = 1’b0 (Disable 100BASE-FX mode)
- Write RDB_Register, offset 0x235, bit[1] = 1’b0 (Disable SGMII-Slave Mode)

Copper(10/100/1000BASE-T) and SGMII-Slave auto-detection is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b1 (Enable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:1] = 2’b01 (Enable Fiber mode)
- Write RDB_Register, offset 0x233, bit[0] = 1’b0 (Disable 100BASE-FX mode)
- Write RDB_Register, offset 0x235, bit[1] = 1’b1 (Enable SGMII-Slave Mode)

Copper(10/100/1000BASE-T) and 1000BASE-X/SGMII-Slave auto-detection is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b1 (Enable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:1] = 2’b01 (Enable Fiber mode)
- Write RDB_Register, offset 0x233, bit[0] = 1’b0 (Disable 100BASE-FX mode)
- Write RDB_Register, offset 0x235, bit[0] = 1’b1 (Enable SGMII-Slave/1000BASE-X Auto-Detect Mode)

1000BASE-X and SGMII-Slave auto-detection is enabled by:

- Write RDB_Register, offset 0x23E, bit[0] = 1’b0 (Disable Copper/Fiber Auto-detection)
- Write RDB_Register, offset 0x021, bits[2:0] = 3’b010 (Enable Fiber mode and access Copper Registers)
- Write Register 0x0, bit[11] = 1’b1 (Power down the Copper interface)
- Write RDB_Register, offset 0x233, bit[0] = 1’b0 (Disable 100BASE-FX mode)
- Write RDB_Register, offset 0x235, bit[0] = 1’b1 (Enable SGMII-Slave/1000BASE-X Auto-Detect Mode)

Signal Detect Mode

When in 100BASE-FX or 1000BASE-X mode the link can be qualified by the SD or RX-LOS signal from the optical module. The SD or RX-LOS signal from the optical module must be connected to the PHY Ports LED[3] input. The link will only come up if Sync_Status is true AND SD/RX-LOS is active. Enabling this mode prevents false links due to noise coming from the optical modules when no fiber cable is connected or noise on the PHY's receiver. To enable Signal Detect Mode do the following:

- Write RDB_Register, offset 0x237, bit[5] = 1'b1 (enable signal detect on LED[3]).
- Set the polarity of the Fiber Signal Detect bit.
  - If the optical module outputs a SD signal, then set RDB_Register, offset 0x235, bit[8] = 1'b0.
  - Write RDB_Register, offset 0xA14, bit[0] = 1'b1 (disable output on LED[3] for all ports).
  - If the optical module outputs a RX_LOS signal, then set RDB_Register, offset 0x235, bit[8] = 1'b1.

Unidirectional Mode

The PHY supports fiber unidirectional mode when in 1000BASE-X or 100BASE-FX mode. The system designer may wish to enable fiber unidirectional mode after the link is established, for example sending 'dying gasp' for OAM or before the link is established. To enable the PHY for fiber unidirectional mode, do the following register writes:

- Write FIBER_CONTROL register 0x00, bits[15:0] = 0x0140 (disable 1000BASE-X auto-negotiation).
- Write FIBER_CONTROL register 0x00, bits[15:0] = 0x2100 (disable 100BASE-FX auto-negotiation).
- Write RDB_Register, offset 0x236.
  - Bit[5] = 1'b0 (disable AMP_SIG_DET_EN)
  - Bit[3] = 1'b0 (disable FILTER_FORCED_LINK)
  - Bit[2] = 1'b0 (disable FALSE_LINK_DIS)
  - Bit[1] = 1'b0 (disable SERDES_AUTONEG_PARALLEL_DET_EN)
- Write FIBER_CONTROL register 0x00, bit[5] = 1'b1 (enable UNIDIRECTIONAL_EN).

To disable fiber unidirectional mode do the following register writes:

- Write FIBER_CONTROL register 0x00, bit[5] = 1'b0 (Disable UNIDIRECTIONAL_EN).
- Write RDB_Register, offset 0x236.
  - Bit[5] = 1'b1 (enable AMP_SIG_DET_EN)
  - Bit[3] = 1'b1 (enable FILTER_FORCED_LINK)
  - Bit[2] = 1'b1 (enable FALSE_LINK_DIS)
  - Bit[1] = 1'b1 (enable SERDES_AUTONEG_PARALLEL_DET_EN)
- Write register 0x00, bits[15:0] to desired speed, duplex and force or auto-negotiation mode.
- If using auto-negotiation mode, restart auto-negotiation (FIBER_CONTROL register 0x00, bit[12] = 1'b1).

Far-End Fault

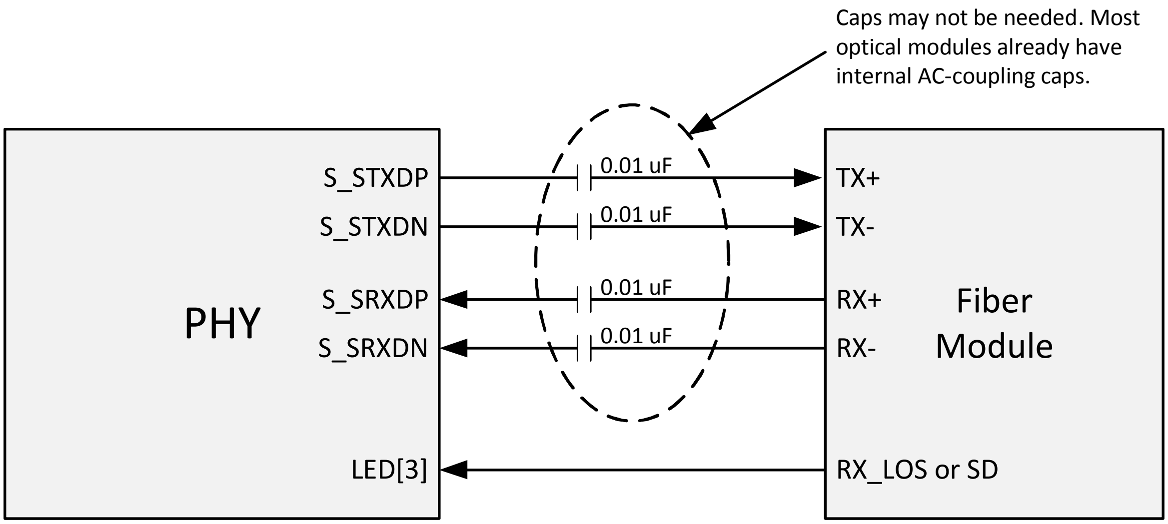
Far-End Fault per IEEE 802.3 section 24.3.2.1 is only supported in 100BASE-FX mode. To enable Far-End Fault do the following:

- RDB_Register, offset 0x037, bit[0] = 1’b1

Fiber Layout

The use of high-speed signal routing techniques is recommended when considering system layout and trace routing for the Fiber signals. Figure 8 on page 54 shows the SGMII connection between the switch and PHY. Broadcom recommends the following:

- The Fiber interface trace routing between the BCM54285 and the Link Partner should use a characteristic differential impedance of 100Ω ±10% and be referenced to a ground plane to increase noise immunity and insure low reflections. To calculate the differential impedance, a 2D field solver is recommended.
- The differential traces should be routed next to each other and the trace lengths should be matched as closely as possible. Avoid matching lengths by adding additional trace or serpentines.
- The differential traces should be routed on one layer if possible. If this is not possible, then make sure that ground vias are placed next to the signal vias.
- The Fiber output data signals from the BCM54285 to the Link Partner may need to be AC coupled typically with a 0.01µF capacitor. Typically SFP modules have internal AC caps.
- The Fiber input data signals from the Link Partner to the BCM54285 needs to be AC coupled typically with a 0.01µF to 0.1µF capacitor.
- The capacitors can be placed any where along the trace as long as the capacitors are equidistant from the BCM54285.
- The receive, clock and transmit signals should be kept away from each other and other analog and clock signals to reduce cross-talk. As a rule of thumb, do not space differential pairs or single-ended traces closer than three times the height above the nearest plane.



- Use 100 ohm characteristic differential impedance for fiber signal traces.
- Use external AC caps (0.01 uF to 0.1 uF) only if the optical module does not have internal AC caps.
- Caps can be placed anywhere as long as they are equidistant from the PHY.
- The fiber inputs have on-chip internal 80 to 120 ohm differential termination.
- RX_LOS or SD signal is optional.

Figure 8: Fiber Interface

Management Interface

The BCM54285 contains a large set of PHY registers. These registers are accessible through the MDIO and MDC serial interface. The functional and electrical interface complies with IEEE Std 802.3, Section 22, and also supports MDC clock rates of up to 12.5 MHz. The management interface supports the defined Status and Control registers of IEEE Std 802.3, Clauses 22, 28, 37, 40, and 45. In addition, the BCM54285 contains multipurpose registers for extended software control.

Caution! The BCM54285's MDIO/MDC will respond to Clause 45 transactions. If a Clause 45 device is on the same MDIO/MDC interface, make sure the Clause 45 devices's PHY address does not overlap with any of the BCM54285's PHY addresses.

MDIO/MDC Access

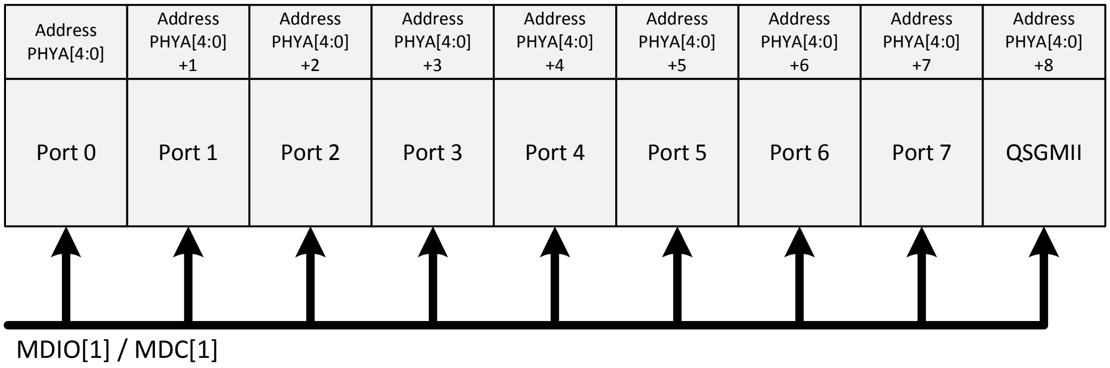
The BCM54285 has eight unique PHY addresses and one OSGMII PHY address for MDIO MII management. The default addresses depend on PHYA[4:0], MDIO_SEL[2:1] and PHYA_REV. Table 12 and Table 13 on page 57 show the various MDIO accesses modes that are supported.

Table 12: MDIO_SEL[2:1] = 00

MDIO_SEL[2:1]	*PHYA_REV*	*Description*
00	0	MDIO[1]/MDC[1]: Port 0 to port 7 and QSGMII. MDIO[2]/MDC[2]: Not used. • Port 0 PHY address = PHYA[4:0] • Port 1 PHY address = PHYA[4:0] + 1 • Port 2 PHY address = PHYA[4:0] + 2 • Port 3 PHY address = PHYA[4:0] + 3 • Port 4 PHY address = PHYA[4:0] + 4 • Port 5 PHY address = PHYA[4:0] + 5 • Port 6 PHY address = PHYA[4:0] + 6 • Port 7 PHY address = PHYA[4:0] + 7 • QSGMII address = PHYA[4:0] + 8 See Figure 9 on page 56.
00	1	MDIO[1]/MDC[1]: Port 0 to port 7 and QSGMII. MDIO[2]/MDC[2]: Not used. • Port 0 PHY address = PHYA[4:0]+ 7 • Port 1 PHY address = PHYA[4:0] + 6 • Port 2 PHY address = PHYA[4:0] + 5 • Port 3 PHY address = PHYA[4:0] + 4 • Port 4 PHY address = PHYA[4:0] + 3 • Port 5 PHY address = PHYA[4:0] + 2 • Port 6 PHY address = PHYA[4:0] + 1 • Port 7 PHY address = PHYA[4:0] • QSGMII address = PHYA[4:0] + 8 See Figure 9 on page 56.

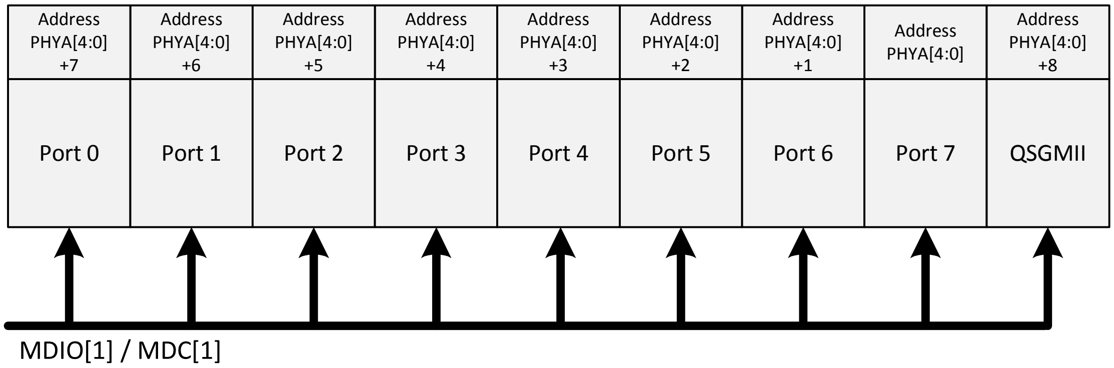
Note: The QSGMII address can be programmed to share Port 3’s PHY address (PHYA[4:0] + 3) by setting RDB_Register, offset 0x810, bit[3:2] = 2’b11. Since accesses to the QSGMII will be minimal, this option will save one PHY address.

MDIO_SEL[2:1] = 00, PHYA_REV = 0



Address PHYA[4:0]	Address PHYA[4:0] +1	Address PHYA[4:0] +2	Address PHYA[4:0] +3	Address PHYA[4:0] +4	Address PHYA[4:0] +5	Address PHYA[4:0] +6	Address PHYA[4:0] +7	Address PHYA[4:0] +8
Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	QSGMII

MDIO_SEL[2:1] = 00, PHYA_REV = 1



Address PHYA[4:0] +7	Address PHYA[4:0] +6	Address PHYA[4:0] +5	Address PHYA[4:0] +4	Address PHYA[4:0] +3	Address PHYA[4:0] +2	Address PHYA[4:0] +1	Address PHYA[4:0]	Address PHYA[4:0] +8
Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	QSGMII

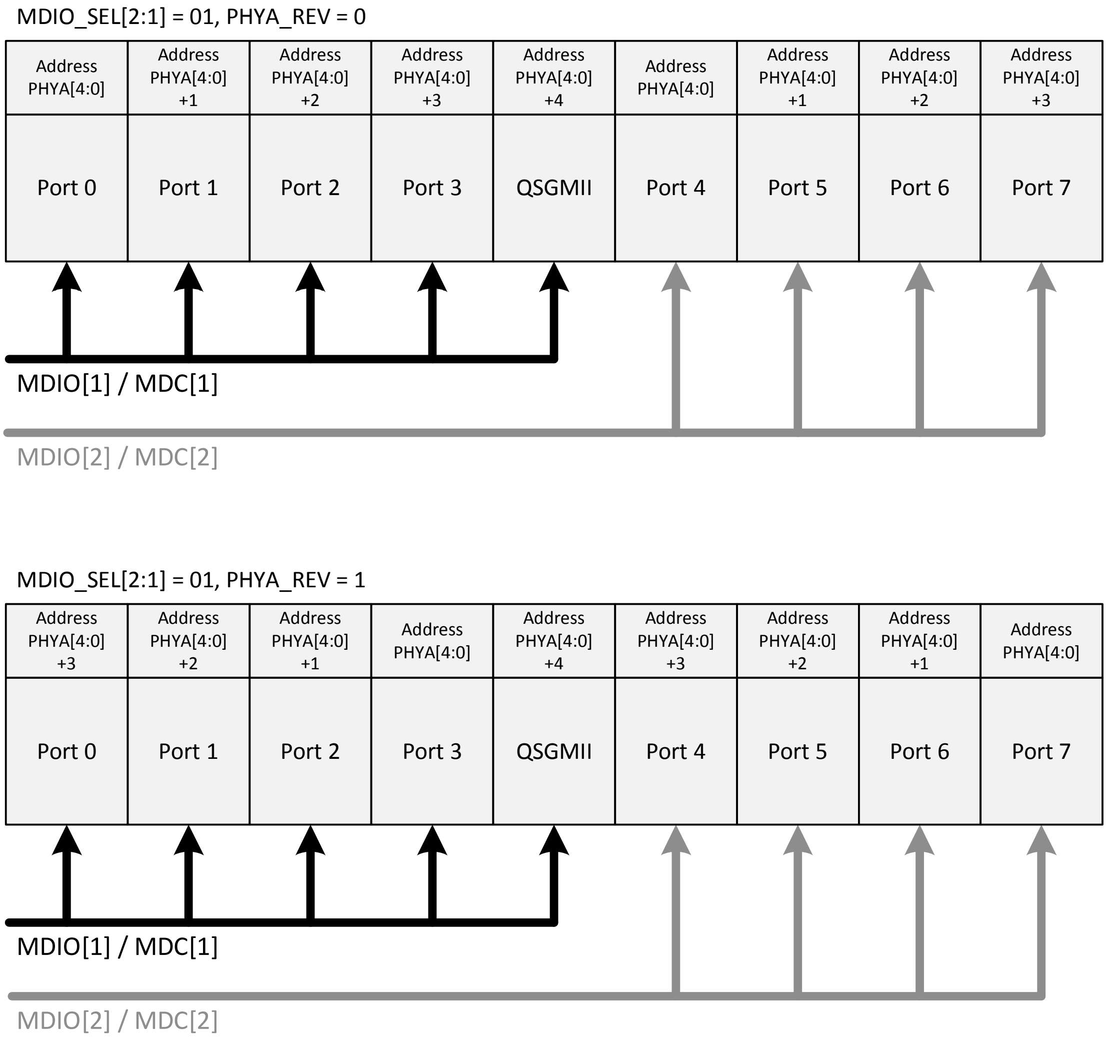
Note: The QSGMII PHY address can optionally be programmed to share Port 3's PHY address.

Figure 9: MDIO_SEL[2:1] = 2'b00

Table 13: MDIO_SEL[2:1] = 01

MDIO_SEL[2:1]	***PHYA_REV***	***Description***
01	0	MDIO[1]/MDC[1]: Port 0 to port 3 and QSGMII. • Port 0 PHY address = PHYA[4:0] • Port 1 PHY address = PHYA[4:0] + 1 • Port 2 PHY address = PHYA[4:0] + 2 • Port 3 PHY address = PHYA[4:0] + 3 • QSGMII address = PHYA[4:0] + 4 MDIO[2]/MDC[2]: Port 4 to port 7. • Port 4 PHY address = PHYA[4:0] • Port 5 PHY address = PHYA[4:0] + 1 • Port 6 PHY address = PHYA[4:0] + 2 • Port 7 PHY address = PHYA[4:0] + 3 See Figure 10 on page 58.
01	1	MDIO[1]/MDC[1]: Port 0 to port 7 and QSGMII. • Port 0 PHY address = PHYA[4:0] +3 • Port 1 PHY address = PHYA[4:0] + 2 • Port 2 PHY address = PHYA[4:0] + 1 • Port 3 PHY address = PHYA[4:0] • QSGMII address = PHYA[4:0] + 4 MDIO[2]/MDC[2]: Port 4 to port 7. • Port 4 PHY address = PHYA[4:0]+ 3 • Port 5 PHY address = PHYA[4:0] + 2 • Port 6 PHY address = PHYA[4:0] + 1 • Port 7 PHY address = PHYA[4:0] See Figure 10 on page 58.

Note: The QSGMII address can be programmed to share Port 3’s PHY address (PHYA[4:0] + 4) by setting RDB_Register, offset 0x810, bit[3:2] = 2’b11. Since accesses to the QSGMII will be minimal, this option will save one PHY address.



Note: The QSGMII PHY address can optionally be programmed to share Port 3's PHY address.

Figure 10: MDIO_SEL[2:1] = 2'b01

Interrupt Interface

The BCM54285 supports three types of interrupt sources and three types of interrupt outputs. Figure 11 shows a simplified interrupt block diagram.

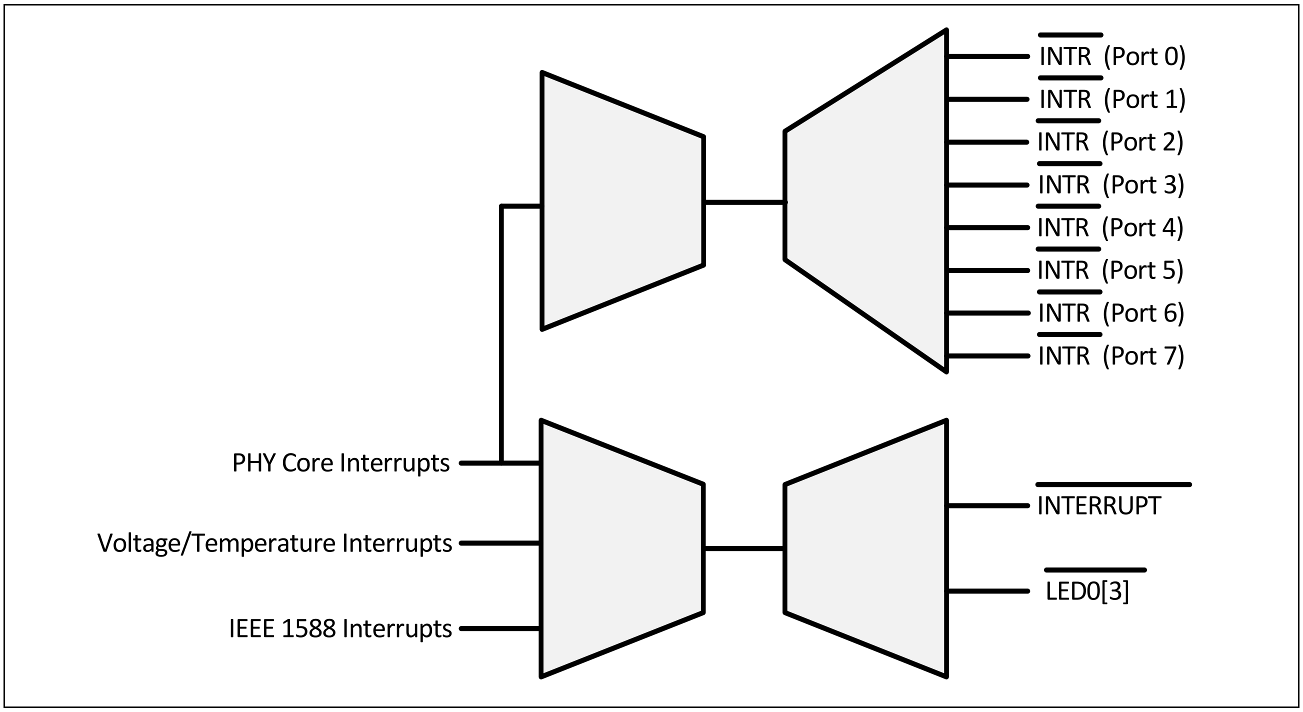


Figure 11: Simplified Interrupt Block Diagram

Interrupt Sources

- PHY Core interrupts
- Voltage/Temperature monitor interrupts
- 1588v2 interrupts

Interrupt Outputs

- Interrupts on any LED programmed to $\overline{\text{INTR}}$ mode.
  - Only PHY Core interrupts are supported.
  - Enabled on a per port basis
- Interrupts on the $\overline{\text{INTERRUPT}}$.
  - Ability to enable or disable any of the following interrupt sources: PHY Core, Voltage/Temperature, or 1588 interrupts.
  - Global interrupt for all ports.
- Interrupts on Port 0's $\overline{\text{LED0[3]}}$ when programmed to $\overline{\text{INTERRUPT}}$ mode.
  Behaves exactly the same way as the $\overline{\text{INTERRUPT}}$.

PHY Core Interrupts

PHY Core interrupts can be programmed to be outputted to the following balls:

- LED programmed to INTR
- INTERRUPT
- LED0[3]: (Port 0, LED[3]) programmed to INTERRUPT mode.

To program the PHY core interrupts on LED programmed to INTR, do the following register writes:

- RDB_Register, offset 0x01D, bits[3:0] = 0x6 (Enable INTR on LED[0])
- RDB_Register, offset 0x01D, bits[7:4] = 0x6 (Enable INTR on LED[1])
- RDB_Register, offset 0x01E, bits[3:0] = 0x6 (Enable INTR on LED[2])
- RDB_Register, offset 0x01E, bits[7:4] = 0x6 (Enable INTR on LED[3])

Note: Only one LED per port should be programmed for the INTR, since all the LEDs for that port will behave exactly the same.

Note: Interrupts are done on a per port basis when INTR is used.

To program the PHY Core interrupts on the INTERRUPT, do the following register writes:

- RDB_Register, offset 0x82D, bit[4] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 0)
- RDB_Register, offset 0x82D, bit[5] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 1)
- RDB_Register, offset 0x82D, bit[6] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 2)
- RDB_Register, offset 0x82D, bit[7] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 3)
- RDB_Register, offset 0x82D, bit[8] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 4)
- RDB_Register, offset 0x82D, bit[9] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 5)
- RDB_Register, offset 0x82D, bit[10] = 1’b01 (Enable PHY Core Interrupts on INTERRUPT for Port 6)
- RDB_Register, offset 0x82D, bit[11] = 1’b0 (Enable PHY Core Interrupts on INTERRUPT for Port 7)

To program the PHY Core interrupts on the LED0[3], do the following register writes:

- RDB_Register, offset 0x811, bit[7] = 1’b1 (Enable INTERRUPT mode on LED0[3])
- RDB_Register, offset 0x82E, bit[4] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 0)
- RDB_Register, offset 0x82E, bit[5] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 1)
- RDB_Register, offset 0x82E, bit[6] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 2)
- RDB_Register, offset 0x82E, bit[7] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 3)
- RDB_Register, offset 0x82E, bit[8] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 4)
- RDB_Register, offset 0x802E, bit[9] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 5)
- RDB_Register, offset 0x82E, bit[10] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 6)
- RDB_Register, offset 0x82E, bit[11] = 1’b0 (Enable PHY Core Interrupts on LED0[3] for Port 7)

Determining/Clearing PHY Core Interrupt

The port that generated the interrupt can be determined by reading RDB_Register, offset 0x03B, bits[7:0].

- Bit[7] = 1’b1: Interrupt generated on Port 7.
- Bit[6] = 1’b1: Interrupt generated on Port 6.
- Bit[5] = 1’b1: Interrupt generated on Port 5.
- Bit[4] = 1’b1: Interrupt generated on Port 4.
- Bit[3] = 1’b1: Interrupt generated on Port 3.
- Bit[2] = 1’b1: Interrupt generated on Port 2.
- Bit[1] = 1’b1: Interrupt generated on Port 1.
- Bit[0] = 1’b1: Interrupt generated on Port 0.

The cause of the interrupt(s) and the clearing of the interrupt(s) is done by reading the following registers:

- COPPER_INTERRUPT_STATUS register: RDB_Register, offset 0x00A
- EXPANSION_INTERRUPT_STATUS register: RDB_Register, offset 0x031, bit[15]

Voltage/Temperature Monitor Interrupts

Table 14 shows the voltage/temperature interrupt mask registers. Voltage/Temperature interrupts can be programmed to be outputted on the INTERRUPT, and/or LED0[3] programmed to INTERRUPT mode. Interrupt output on LEDs programmed to INTR are not supported.

To program the voltage/temperature interrupts on the INTERRUPT, do the following register writes:

- RDB_Register, offset 0x82D, bit[3] = 1’b0 (enable voltage/temperature interrupts on INTERRUPT

To program the voltage/temperature interrupts on the LED0[3], do the following register writes:

- RDB_Register, offset 0x811, bit[7] = 1’b1 (enable INTERRUPT mode on LED0[3])
- RDB_Register, offset 0x82E, bit[3] = 1’b0 (enable voltage/temperature monitor interrupts)

Determining/Clearing Voltage/Temperature Interrupts

The cause of the interrupt(s) and the clearing of the interrupt(s) is done by reading the VOLT_TEMP_MONITOR register: RDB_Register, offset 0x83B, bits[2:0]

Table 14: Voltage/Temperature Interrupts

Interrupt Name	*Register*	*Comments*
VTMON_V3P3_2P5_INTR_MASK	VOLT_TEMP_MONITOR register RDB_Register, offset 0x83B, bit[10] VOLT_TEMP_MONITOR register RDB_Register, offset 0x83B, bit[2]	Voltage supply 3.3V/2.5V monitor interrupt. Voltage is out of threshold settings since last read. High threshold is set by RDB_Register, offset 0x8039, bits[9:0]. Low threshold is set by RDB_Register, offset 0x83A, bits[9:0].

Table 14: Voltage/Temperature Interrupts (Cont.)

Interrupt Name	*Register*	*Comments*
VTMON_V1P0_INTR_MASK	VOLT_TEMP_MONITOR register RDB_Register, offset 0x83B, bit[9] VOLT_TEMP_MONITOR register RDB_Register, offset 0x83B, bit[1]	Voltage supply 1.0V monitor interrupt. Voltage is out of threshold settings since last read. • High threshold is set by RDB_Register, offset 0x836, bits[9:0]. • Low threshold is set by RDB_Register, offset 0x837, bits[9:0].
VTMON_TEMP_INTR_MASK	VOLT_TEMP_MONITOR RDB_Register, offset 0x83B, bit[8] VOLT_TEMP_MONITOR register RDB_Register, offset 0x83B, bit[0]	Temperature is out of threshold settings since last read. • High threshold is set by RDB_Register, offset 0x833, bits[9:0]. • Low threshold is set by RDB_Register, offset 0x834, bits[9:0].

1588v2 Interrupts

1588v2 interrupts can be programmed to be outputted on the INTERRUPT, and/or LED0[3] programmed to INTERRUPT mode. Interrupt output on LEDs programmed to INTR are not supported.

To program the 1588v2 interrupts on the INTERRUPT, do the following register writes:

- RDB_Register, offset 0x82D, bit[0] = 1’b1 (enable 1588v2 Interrupts on INTERRUPT)

To program the 1588v2 interrupts on the LED0[3], do the following register writes:

- RDB_Register, offset 0x811, bit[7] = 1’b1 (enable INTERRUPT mode on LED0[3])
- RDB_Register, offset 0x82E, bit[0] = 1’b1 (enable 1588v2 Interrupts)

Determining/Clearing 1588v2 Interrupts

The cause of the interrupt(s) and the clearing of the interrupt(s) is done by reading the P1588_INTERRUPT_STATUS register: RDB_Register, offset 0xA5F, bits[1:0]

LED Interface

The BCM54285 supports the following LED modes:

- “Parallel LED Mode” on page 63
- “External Serial LED Interface” on page 63
- “Constant Current LED Mode” on page 68
- “Legacy Constant Current LED Mode” on page 68

Each Port supports up to 4 LEDs. LED[0], LED[1], LED[2], and LED[3].

Parallel LED Mode

The BCM54285 has four programmable LED balls per port that perform different functions. Each of the BCM54285 LEDs can be individually programmed to any one of the many LED modes on a per port basis. The available LED modes are:

- LINKSPD1
- LINKSPD2
- XMITLED
- ACTIVITY
- FDXLED
- SLAVE
- INTR
- QUALITY
- RCVLED
- WIRESPEED DOWNGRADE
- MULTICOLOR[1]
- MULTICOLOR[2]
- CABLE DIAG (Open/Short found)
- ON
- OFF

Any LED programmed for the above modes, except for INTR LED, behave as active low push-pull outputs. Any LED programmed to INTR behaves as an active low open-drain output. The LEDs can be programmed through RDB_Register, offset 0x01D and RDB_Register, offset 0x01E.

External Serial LED Interface

External Serial LED mode provides a means for system vendors to control the BCM54285's LED behavior directly by providing a LED serial stream to the BCM54285. This serial LED stream is strobed into and out of the chip with a serial clock that may run up to 25 MHz. A strobe signal is provided to asynchronously shift data from the serializer to the LED output buffers on a rising edge. Figure 12 on page 64 shows the Serial LED block diagram. Table 15 on page 65 shows the External Serial LED hardware balls and Table 16 on page 65 shows the software bits used for External Serial LED mode. The External Serial LEDs are mapped to the PHY's Port LEDs as shown in Table 17 to Table 20. Even though the Serial LEDs are mapped to the physical Port 0 to Port 7 LEDs, the incoming serial stream does not have to be mapped to the Port 0 to Port 7 LEDs. It is up to the customer to determine what the serial stream will be.

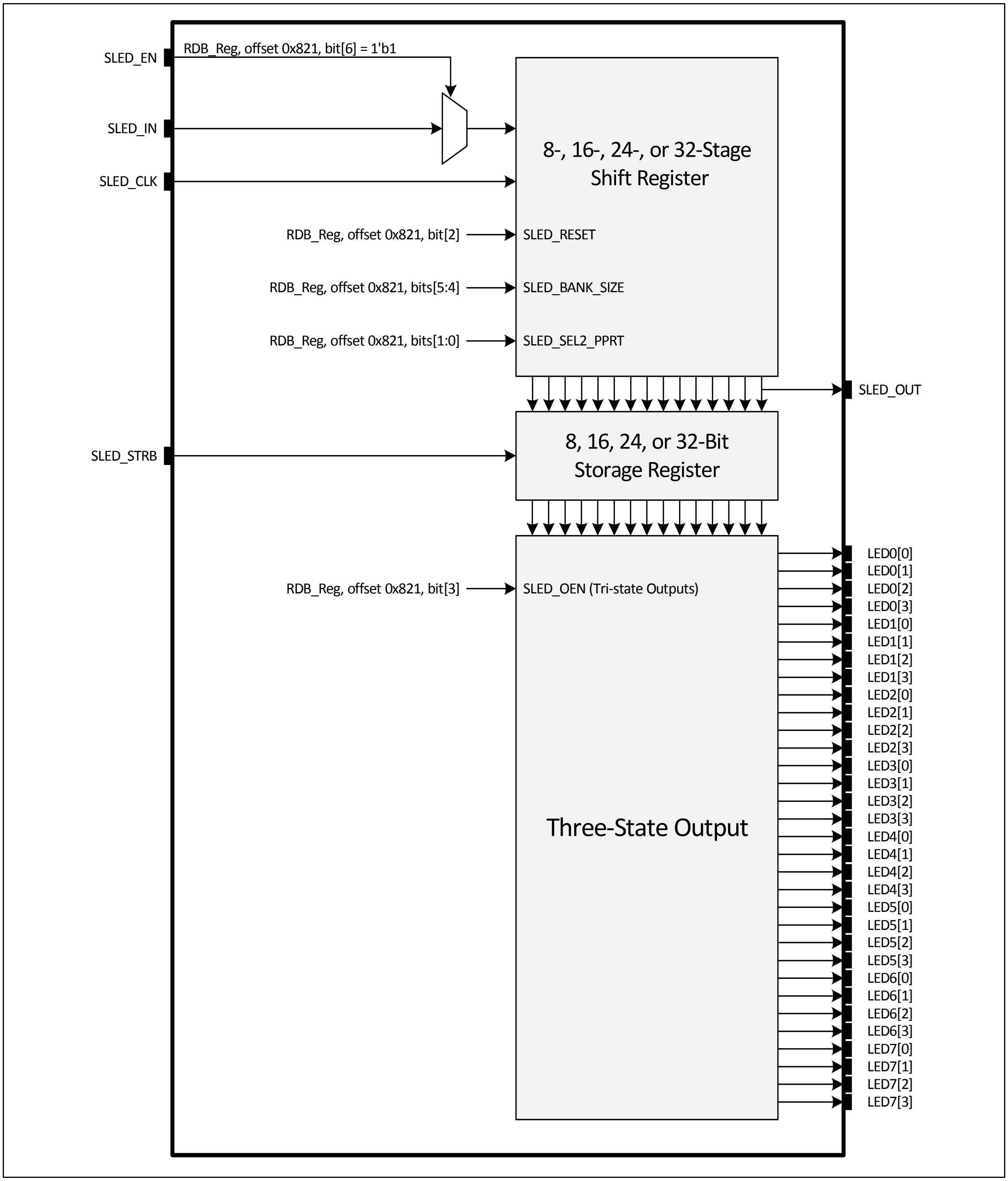


Figure 12: External Serial LED Block Diagram

Table 15: External Serial LED Hardware Balls

Ball Name	***I/O***	***Function***
SLED_EN	I	External serial LED enable. External serial LED mode is enabled when this ball is pulled-high at power-up. After power-up: • LED[0] and LED[2] from ports 0 to 7 will be driven low. • LED[1] and LED[3] from ports 0 to 7 will be driven high.
SLED_IN	I	External serial LED stream input.
SLED_CLK	I	External serial LED clock input. This clock can be run up to 25 MHz.
SLED_STRB	I	External serial LED strobe signal used to shift the contents onto the LED outputs.
SLED_OUT	O	External serial LED output ball that can be daisy-chained to the next PHY’s SLED_IN input.
LEDx_[3:0]	O	External serial LEDs. These are the same LEDs used for parallel LED mode. ***Note:*** x = Port number. Port numbers range from 0 to 7.‘

⚠ **Caution!** During hardware or software reset, the following balls must be pulled low: SLED_IN, SLED_CLK and SLED_STRB.

Table 16: External Serial LED Software Bits

Bit Name	***Function***
Enable External Serial LED mode	External serial LED mode is enabled when RDB_Register, offset 0x821, bit[6] = 1’b1.
Reset External Serial LED mode	When RDB_Register, offset 0x821, bit[2] = 1’b1: • LED[0] and LED[2] from ports 0 to 7 will be driven low. • LED[1] and LED[3] from ports 0 to 7 will be driven high. • Contents of the internal shift register will be set to zeros. ***Note:*** This bit is not self-clearing and must be set = 1’b0 for normal operation.
Tristate External Serial LED outputs	The serial LED outputs will be tristated when RDB_Register, offset 0x821, bit[3] = 1’b1.
SLED_BANK_SIZE	RDB_Register, offset 0x821, bits[5:4] controls which LEDs to use. • Bits[5:4] = 2’b00: Use four LEDs per port (LEDx[3:0]). • Bits[5:4] = 2’b01: Use three LEDs per port.(LEDx[2:0]) • Bits[5:4] = 2’b10: Use two LEDs per port.(LEDx[1:0]) • Bits[5:4] = 2’b11: Use one LEDs per port.(LEDx[0]) ***Note:*** x = Port number.
SLED_SEL2_PPRT	RDB_Register, offset 0x821, bits[1:0] controls which port LEDs to use. • Bits[5:4] = 2’b00: Port 0 to port 7 LEDs used. • Bits[5:4] = 2’b01: Port 0 to port 3 LEDs used. • Bits[5:4] = 2’b10: Port 0 to port 1 LEDs used. • Bits[5:4] = 2’b11: Port 0 LEDs used.

Table 17: External Serial LED Select (Ports 0 to 7)

SLED_SEL2_PPRT ***RDB_Register, offset 0x821, bits[1:0]***	***SLED_BANK_SIZE*** ***RDB_Register, offset 0x821, bits[5:4]***	***LEDs Selected***	***Comments***
00	00	LED0_[3:0] LED1_[3:0] LED2_[3:0] LED3_[3:0] LED4_[3:0] LED5_[3:0] LED6_[3:0] LED7_[3:0]	Thirty two LEDs (four LEDs per port). LED[0], LED[1], LED[2], and LED[4] selected for ports 0 to 7.
00	01	LED0_[2:0] LED1_[2:0] LED2_[2:0] LED3_[2:0] LED4_[2:0] LED5_[2:0] LED6_[2:0] LED7_[2:0]	Twenty four LEDs (three LEDs per port). LED[0], LED[1], and LED[2] selected for ports 0 to 7.
00	10	LED0_[1:0] LED1_[1:0] LED2_[1:0] LED3_[1:0] LED4_[1:0] LED5_[1:0] LED6_[1:0] LED7_[1:0]	Sixteen LEDs (two LEDs per port). LED[0] and LED[1] selected for ports 0 to 7.
00	11	LED0_[0] LED1_[0] LED2_[0] LED3_[0] LED4_[0] LED5_[0] LED6_[0] LED7_[0]	Eight LEDs (one LED per port). LED[0] selected for ports 0 to 7.

Table 18: External Serial LED Select (Ports 0 to 3)

SLED_SEL2_PPRT ***RDB_Register, offset 0x821, bits[1:0]***	***SLED_BANK_SIZE*** ***RDB_Register, offset 0x821, bits[5:4]***	***LEDs Selected***	***Comments***
01	00	LED0_[3:0] LED1_[3:0] LED2_[3:0] LED3_[3:0]	Sixteen LEDs (four LEDs per port). LED[0], LED[1], LED[2] and LED[4] selected for ports 0 to 3.
01	01	LED0_[2:0] LED1_[2:0] LED2_[2:0] LED3_[2:0]	Twelve LEDs (three LEDs per port). LED[0], LED[1], and LED[2] selected for ports 0 to 3.
01	10	LED0_[1:0] LED1_[1:0] LED2_[1:0] LED3_[1:0]	Eight LEDs (two LEDs per port.) LED[0] and LED[1] selected for ports 0 to 3.
01	11	LED0_[0] LED1_[0] LED2_[0] LED3_[0]	Four LEDs (one LED per port.) LED[0] selected for ports 0 to 3.

Table 19: External Serial LED Select (Ports 0 to 1)

SLED_SEL2_PPRT ***RDB_Register, offset 0x821, bits[1:0]***	***SLED_BANK_SIZE*** ***RDB_Register, offset 0x821, bits[5:4]***	***LEDs Selected***	***Comments***
01	00	LED0_[3:0] LED1_[3:0]	Eight LEDs (four LEDs per port). LED[0], LED[1], LED[2] and LED[4] selected for ports 0 to 1.
01	01	LED0_[2:0] LED1_[2:0]	Six LEDs (three LEDs per port). LED[0], LED[1], and LED[2] selected for ports 0 to 1.
01	10	LED0_[1:0] LED1_[1:0]	Four LEDs (two LEDs per port.) LED[0] and LED[1] selected for ports 0 to 1.
01	11	LED0_[0] LED1_[0]	Two LEDs (one LED per port.) LED[0] selected for ports 0 to 1.

Table 20: External Serial LED Select (Port 0)

SLED_SEL2_PPRT RDB_Register, offset 0x821, bits[1:0]	*SLED_BANK_SIZE RDB_Register, offset 0x821, bits[5:4]*	*LEDs Selected*	*Comments*
11	00	LED0_[3:0]	Four LEDs (four LEDs per port). LED[0], LED[1], LED[2] and LED[4] selected for port 0.
11	01	LED0_[2:0]	Three LEDs (three LEDs per port). LED[0], LED[1], and LED[2] selected for port 0.
11	10	LED0_[1:0]	Two LEDs (two LEDs per port.) LED[0] and LED[1] selected for port 0.
11	11	LED0_[0]	One LED (one LED per port.) LED[0] selected for port 0.

Constant Current LED Mode

Constant Current LED mode allows the LED output balls to output a constant current ranging from 2 mA to 16 mA in 2 mA increments. There is no need for external series resistors since the amount of current through the LEDs is programmable. The amount of current is set by the LED_PROGRAMMABLE_CURRENT_MODE_CONTROL register (RDB_Register, offset 0x074, bits[15:0]).

Constant Current LED Mode can be enabled for all ports through hardware, by setting the CRNT_LED_EN = OVDD. Constant Current LED Mode can be enabled or disabled on an individual LED basis by writing to RDB_Register, offset 0x01F, bits[3:0] for each port. Bit[3] controls LED[3], bit[2] controls LED[2], bit[1] controls LED[1], and bit[0] controls LED[0].

Note: Constant Current LED mode cannot be used in conjunction with External Serial LED mode.

Legacy Constant Current LED Mode

Legacy Constant Current LED Mode allows the LEDs programmed in Parallel LED mode to output a constant current ranging from 2 mA to 16 mA in 2 mA increments Legacy Constant Current LED mode. The amount of current is set by RDB_Register, offset 0x074, bits[15:0].

Legacy Constant Current LED Mode is enabled by setting the CRNT_LED_EN = OVDD and setting RDB_Register, offset 0x820, bit[15] = 1’b1. Legacy Constant Current LED Mode can be disabled on an individual LED basis by writing to RDB_Register, offset 0x01F, bits[3:0] for each port. Bit[3] controls LED[3], bit[2] controls LED[2], bit[1] controls LED[1], and bit[0] controls LED[0].

Note: Legacy Constant Current LED Mode cannot be used for LEDs programmed for $\overline{\text{INTR}}$.

If the $\overline{\text{INTR}}$ function is needed, then the LED programmed for $\overline{\text{INTR}}$ must have Constant Current LED Mode disabled by writing to RDB_Register, offset 0x01F, bits[3:0].

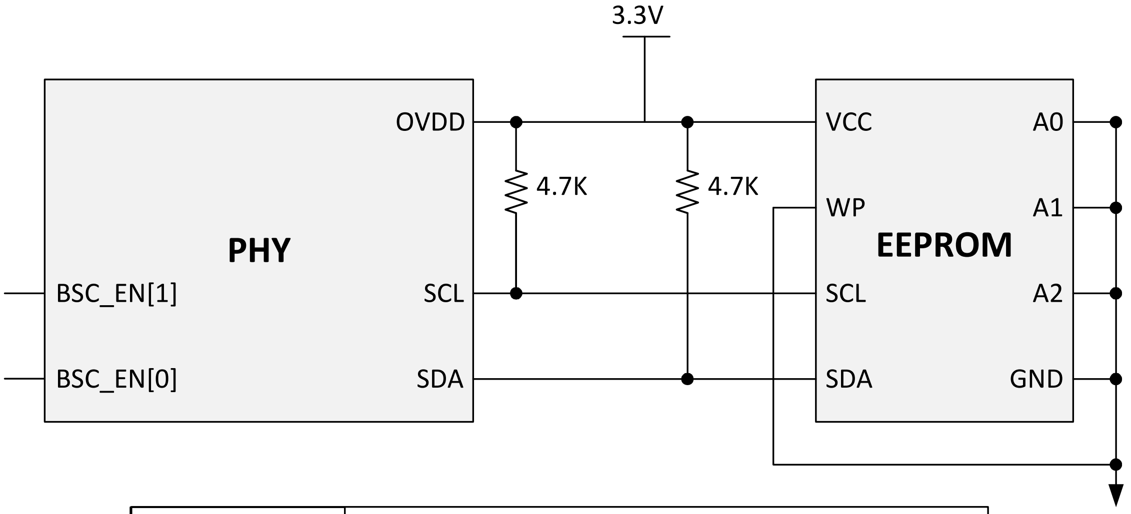
Broadcom Serial Control Interface

Overview

The BCM54285's registers can be programmed through an external EEPROM through the BCM54285's Broadcom Serial Control Interface (BSC) interface. The BSC interface is compatible with the Phillips® (now NXP) I^{2}C bus interface operating at 100 Kbps. This feature is useful in unmanaged systems.

The BCM54285's BSC interface is a Master device that uses a two wire (Serial Clock and Serial Data) open collector interface. The Serial Clock (SCL) is an output from the BCM54285 and operates at 100 kHz. The Serial Data (SDA) is bidirectional.

Although BSC protocol supports multiple master designs, the BCM54285 interface supports only a single master implementation. Each BCM54285 must have its own EEPROM. Figure 13 on page 70 shows a typical connection between the BCM54285 and an EEPROM.



BSC_EN[1:0]	Description
00	Disable booting from PROM
11	Enable booting from PROM

Note: The SCL and SDA pins must be pulled high, even if the BSC interface is not used.
The EEPROM address pins (A0, A1, and A2) must be pulled to ground.
The EEPROM write protect pin (WP) must be pulled to ground.

Figure 13: BSC Interface

Section 5: Ball Descriptions

The following conventions are used to identify the I/O types in Table 22 on page 72. The I/O ball type is useful in referencing the DC ball characteristics contained in Section 10: “Electrical Characteristics”.

Table 21: I/O Signal Type Definitions

Abbreviation	***Description***
I_D	LVCMOS input
I_{PU}	Input with Internal pull-up resistor
I_{PD}	Input with Internal pull-down resistor
I_{CS}	Input continuously sampled
I_{SOR}	Input sampled on reset
I_{ST}	Schmitt trigger input
I_{3T}	3.3V tolerant input
I/O	Bidirectional
O	Output
O_T	Tristateable output
O_D	LVCMOS output
O_{OD}	Open drain output
O_{OC}	Open collector output
A	Analog ball type
B	Bias ball type
Q	QSGMII ball type
XYZ	Active low signal
DNC	Do not connect
NC	No connect
PWR	Power ball
GND	Ground ball

Table 22: Ball Descriptions

<table>
<tr><th>Label</th><th>I/O</th><th>Description</th></tr>
<tr><td colspan="3"><i>Twisted-Pair Interface Connection</i>
In 1000BASE-T mode, differential data from the media is transmitted and received on all four signal pairs. The twisted-pair interface uses the following format: TDx_y_z
• TD is the twisted-pair interface.
• x is the polarity. P = Positive, N = Negative
• y is the port number (0 to 7).
• z is the pair (channel) number (0 to 3).
<i>Note:</i> These balls have internal 50Ω-series termination resistors. There is no need for external termination resistors.
<i>Note:</i> Any unused pairs can be left floating.</td></tr>
<tr><td>TDP_0_0, TDN_0_0
TDP_0_1, TDN_0_1
TDP_0_2, TDN_0_2
TDP_0_3, TDN_0_3</td><td>I/O_A</td><td>Port 0 Copper Interface.</td></tr>
<tr><td>TDP_1_0, TDN_1_0
TDP_1_1, TDN_1_1
TDP_1_2, TDN_1_2
TDP_1_3, TDN_1_3</td><td>I/O_A</td><td>Port 1 Copper Interface.</td></tr>
<tr><td>TDP_2_0, TDN_2_0
TDP_2_1, TDN_2_1
TDP_2_2, TDN_2_2
TDP_2_3, TDN_2_3</td><td>I/O_A</td><td>Port 2 Copper Interface.</td></tr>
<tr><td>TDP_3_0, TDN_3_0
TDP_3_1, TDN_3_1
TDP_3_2, TDN_3_2
TDP_3_3, TDN_3_3</td><td>I/O_A</td><td>Port 3 Copper Interface.</td></tr>
<tr><td>TDP_4_0, TDN_4_0
TDP_4_1, TDN_4_1
TDP_4_2, TDN_4_2
TDP_4_3, TDN_4_3</td><td>I/O_A</td><td>Port 4 Copper Interface.</td></tr>
<tr><td>TDP_5_0, TDN_5_0
TDP_5_1, TDN_5_1
TDP_5_2, TDN_5_2
TDP_5_3, TDN_5_3</td><td>I/O_A</td><td>Port 5 Copper Interface.</td></tr>
<tr><td>TDP_6_0, TDN_6_0
TDP_6_1, TDN_6_1
TDP_6_2, TDN_6_2
TDP_6_3, TDN_6_3</td><td>I/O_A</td><td>Port 6 Copper Interface.</td></tr>
</table>

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
TDP_7_0, TDN_7_0 TDP_7_1, TDN_7_1 TDP_7_2, TDN_7_2 TDP_7_3, TDN_7_3	I/O$_A$	**Port 7 Copper Interface.**
Fiber Interface The fiber receiver uses the following format: SRXDy[z] • y is the polarity. P = Positive, N = Negative • [z] is the port number (0 to 7). The fiber transmitter uses the following format: STXDy[z] • y is the polarity. P = Positive, N = Negative • [z] is the port number (0 to 7).		
SRXDN[0], SRXDP[0] SRXDN[1], SRXDP[1] SRXDN[2], SRXDP[2] SRXDN[3], SRXDP[3] SRXDN[4], SRXDP[4] SRXDN[5], SRXDP[5] SRXDN[6], SRXDP[6] SRXDN[7], SRXDP[7]	I$_S$	**Fiber Interface Data Input.** Differential serial data input when the PHY is in the following: • QSGMII-to-SGMII-Slave • QSGMII-to-100BASE-FX • QSGMII-to-1000BASE-X Data flow is from external fiber SFP output or external SGMII Master (SGMII to 10/100/1000BASE-T) SFP output to PHY fiber input. ***Note:*** Each differential pair has an 80Ω to 120Ω differential impedance. It is highly recommended to use 0.01 µF coupling capacitors.
STXDN[0], STXDP[0] STXDN[1], STXDP[1] STXDN[2], STXDP[2] STXDN[3], STXDP[3] STXDN[4], STXDP[4] STXDN[5], STXDP[5] STXDN[6], STXDP[6] STXDN[7], STXDP[7]	O$_S$	**Fiber Interface Data Output.** Differential serial data output when the PHY is in the following: • QSGMII-to-SGMII-Slave • QSGMII-to-100BASE-FX • QSGMII-to-1000BASE-X. Data flow is from PHY fiber output to external fiber SFP input or external SGMII-Master (SGMII to 10/100/1000BASE-T) SFP input.

Table 22: Ball Descriptions (Cont.)

Label	***I/O***	***Description***
QSGMII Interface The QSGMII Receive interface uses the following format: RDyz • y is either the polarity. P = Positive, N = Negative • z is the QSGMII quad number (0 to 1). The QSGMII Transmit interface uses the following format: TDyz • y is either the polarity. P = Positive, N = Negative • z is the QSGMII quad number (0 to 1).		
RDP0, RDN0 RDP1, RDN1	I_Q	**QSGMII Interface Data Input.** Differential serial data input when the BCM54285 is in QSGMII mode. Data flow is from switch's QSGMII output to PHY's QSGMII input. • RDP0 and RDN0 support Ports 0, 1, 2, and 3. • RDP1 and RDN1 support Ports 4, 5, 6, and 7. ***Note:*** Each differential pair has an 80Ω to 120Ω differential impedance. it is highly recommended to use 0.1 µF to 0.01 µF coupling capacitors.
TDP0, TDN0 TDP1, TDN1	O_Q	**QSGMII Interface Data Output.** Differential serial data output when the BCM54285 is in QSGMII mode. Data flow is from PHY's QSGMII output to switch's QSGMII input. • TDP0 and TDN0 supports Ports 0, 1, 2, and 3. • TDP1 and TDN1 supports Ports 4, 5, 6, and 7.
Mode Select ***V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDDMDIO. See*** Section 10: "Electrical Characteristics"**.**		
INTF_SEL[0] INTF_SEL[1]	$I_{PD,SOR,D}$	**Interface Mode Select.** Active high. Select the PHY configuration shown below: INTF-SEL[1:0] Mode 00 QSGMII to copper 01 QSGMII to fiber 10 Reserved 11 Reserved ***Note:*** • Copper = 10BASE-T, 100BASE-TX 1000BASE-T • Fiber = 100BASE-FX, 1000BASE-X, SGMII-Slave

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
MDIO/MDC Interface V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDDMDIO. See Section 10: “Electrical Characteristics”.		
MDIO[1] MDIO[2]	I/O $_{PU, CS, D}$	**Management Data I/O.** This serial management input/output is used to read from and write to the MII registers. The data value is valid and latched on the rising edge of MDC. These balls can operate at either 1.2V, 2.5V or 3.3V depending on the MDIO_LVL and OVDDMDIO settings listed below. • 1.2V operation: MDIO_LVL = 0, OVDDMDIO = 1.2V • 2.5V operation: MDIO_LVL = 1, OVDDMDIO = 2.5V • 3.3V operation: MDIO_LVL = 1, OVDDMDIO = 3.3V
MDC[1] MDC[2]	I $_{PD, CS, ST}$	**Management Data Clock.** The MDC clock input must be provided to allow MII management functions. Clock frequencies of up to 12.5 MHz are supported. These balls can operate at either 1.2V, 2.5V or 3.3V depending on the MDIO_LVL and OVDDMDIO settings listed below. • 1.2V operation: MDIO_LVL = 0, OVDDMDIO = 1.2V • 2.5V operation: MDIO_LVL = 1, OVDDMDIO = 2.5V • 3.3V operation: MDIO_LVL = 1, OVDDMDIO = 3.3V
MDIO/MDC Configuration V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.		
MDIO_LVL	I_{PU}	**MDIO/MDC Supply Level.** Sets the voltage the OVDDMDIO ball can operate at. • MDIO_LVL = 0: OVDDMDIO operates at 1.2V. • MDIO_LVL = 1: OVDDMDIO can operate at either 2.5V or 3.3V.
MDIO_SEL[1] MDIO_SEL[2]	$I_{PD, CS, D}$	**Management Data I/O Select.** Active high. Selects MDIO access method. See “Management Interface” on page 54.
PHYA[0] PHYA[1] PHYA[2] PHYA[3] PHYA[4]	$I_{PD, SOR, D}$	**PHY Address Select.** Active high. Sets the MII management PHY address. See “Management Interface” on page 54.
PHYA_REV	$I_{PD, SOR, D}$	**PHY Address Reverse.** Active high. When pulled high at reset, the PHY address the ports responds to is reversed. See “Management Interface” on page 54.

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
EEPROM Interface ***V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.***		
BSC_EN[0] BSC_EN[1]	$I_{PD, SOR, D}$	**BSC Enable.** Active high. Broadcom Serial Control. The BSC interface acts as an I^2C master. These inputs enable and select the type of EEPROM. See “Broadcom Serial Control Interface” on page 69 for details. BSC_EN[1:0]: 00 = Disable BSC interface. 11 = Auto detect between Small and Large PROM. ***Note:*** The BSC bus is Philips I^2C compatible.
SCL	O_D	**BSC Serial Data Clock.** Open collector output. 100 kHz clock output. An external pull-up resistor to OVDD must be used. ***Note:*** This ball must be pulled high if the BSC interface is not used. ***Note:*** The BSC bus is Philips I^2C compatible.
SDA	I/O_D	**BSC Serial Data line.** I/O, Open collector output. Bidirectional serial data. An external pull-up resistor to OVDD must be used. ***Note:*** This ball must be pulled high if the BSC interface is not used. ***Note:*** The BSC bus is Philips I^2C compatible.
JTAG ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDDJTAG.		
$\overline{\text{TRST}}$	$I_{ST, PD, CS, D}$	**JTAG Reset.** Active-low. Resets the JTAG controller. This input must be pulled low during normal operation.
TDI	$I_{PU, CS, D}$	**JTAG Test Data Input.** Active-high. JTAG serial data input.
TCK	$I_{PU, CS, D}$	**JTAG Test Clock.** Active-high. JTAG serial clock input.
TMS	$I_{PU, CS, D}$	**JTAG Test Mode Select.** Active-high. JTAG mode select input.
TDO	$O_{T, D}$	**JTAG Test Data Output.** JTAG serial data output.
Test ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.		
TEST[0] TEST[1]	$I_{PD, CS, D}$	**Test Mode Enables.** Active high. These balls must always be pulled low during normal operation. Pulling both balls high tristates all outputs. These balls are used by Broadcom for test purposes only. These balls should be tied to ground.
TVCO[1] TVCO[2]	O_A	**Transmit Test Clock.** 125 MHz transmit test clock used for IEEE conformance testing.

Table 22: Ball Descriptions (Cont.)

Label	***I/O***	***Description***
Reference Clock Configuration ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: "Electrical Characteristics".		
SCLK_TYPE[0] SCLK_TYPE[1]	$I_{PD,SOR,D}$	**Reference Clock Type.** Active high. Selects the type of reference clock REFCLKP and REFCLKN will accept. See "Reference Clock Using REFCLKP/N" on page 93. SCLK_TYPE[1:0] • 00 = CML differential clock input on REFCLKP and REFCLKN. • 01 = Single-ended CMOS input on REFCLKP. • 10 = Crystal input on REFCLKP and REFCLKN. • 11 = NA
SCLK_FREQ[0] SCLK_FREQ[1]	$I_{PD,SOR,D}$	**Reference Clock Frequency.** Active high. Selects the frequency the input reference clock. See "Reference Clock Using REFCLKP/N" on page 93. SCLK_FREQ[1:0]: • 00 = 25 MHz • 01 = 125 MHz • 10 = 312.5 MHz • 11 = 156.25 MHz
Reference Clock Inputs ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by CLKVDD. See Section 10: "Electrical Characteristics"		
REFCLKP REFCLKN	I_A	**Reference Clock Inputs.** configure REFCLKP and REFCLKP to accept differential, single-ended or crystal clock sources depending on the configuration of the balls listed below: • SCLK_TYPE[1:0] • SCLK_FREQ[1:0] See "Reference Clock Using REFCLKP/N" on page 93 for details.
Synchronous Ethernet ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: "Electrical Characteristics".		
REC_CLK[1] REC_CLK[2]	O_T	**Recovered Clock Output.** The recovery clock from the line-side link partner can be multiplexed onto the REC_CLK[1] and or REC_CLK[2] for Synchronous Ethernet applications. See "Synchronous Ethernet" on page 32.
LOC_REC_CLK[1] LOC_REC_CLK[2]	O_T	**Copper PLL Reference Clock Lock.** Active high. When high, indicates the recovered clock is locked to the incoming signal. It does not indicate the quality of the recovered clock. See "Synchronous Ethernet" on page 32. • LOC_REC_CLK[1]: Indicates REC_CLK[1] has achieved lock with the incoming signal. • LOC_REC_CLK[2]: Indicates REC_CLK[2] has achieved lock with the incoming signal.

Table 22: Ball Descriptions (Cont.)

Label	I/O	Description
1588 ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.		
SYNC_IN	I_{PD}	**1588 Sync Input.** Accepts external synchronization signals.
SYNC_OUT	I/O_T	**1588 Sync Output.** Generates or accepts synchronization signals. ***Note:*** To enable SYNC_OUT the following register write must be done. Write RDB_Register offset 0x811, bit[4] = 1'b1.
Reset ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.		
$\overline{\text{RESET}}$	$I_{PU, CS, ST, D}$	**Reset.** Active low. Resets the BCM54285. At reset, all internal registers are restored to the default state and reconfigured based on hardware configuration signals.

Table 22: Ball Descriptions (Cont.)

<table>
<tr><th>Label</th><th>I/O</th><th>Description</th></tr>
<tr><td colspan="3">

Twisted-Pair Media Configuration

Note: V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.

</td></tr>
<tr><td>ANEN</td><td>I_{PU}, SOR, D</td><td>

Auto-negotiation Select. Active high. When high, auto-negotiation is enabled; when low, auto-negotiation is disabled. After reset, the auto-negotiation function is under software control.

This ball works in conjunction with the F1000, FDX, and SPD0 to set up the default duplex and speed of the twisted-pair interface.

</td></tr>
<tr><td>SPD0</td><td>I_{PD}, SOR, D</td><td rowspan="2">

Speed Select. Active high. Sets the default advertisement of the BCM54285, according to the following table.

<table>
<tr><th>ANEN</th><th>F1000</th><th>SPD0</th><th>Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Force 10BASE-T</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Force 100BASE-TX</td></tr>
<tr><td>0</td><td>1</td><td>X</td><td>Force 1000BASE-T (See Note below)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Auto-negotiate advertise: 10BASE-T</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Auto-negotiate advertise: 10/100BASE-TX</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Auto-negotiate advertise: 10/100/1000BASE-T</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Auto-negotiate advertise: 1000BASE-T</td></tr>
</table>

Note: FORCE 1000BASE-T mode is for test purposes only, and disabling auto-negotiation can lead to link-configuration mismatches and no-link situations.

The *Annex 28D.5 Extensions Required for Clause 40 (1000BASE-T)* IEEE 802.3 specification requires that auto-negotiation be used in 1000BASE-T operation. There are no standards that govern a protocol for 1000BASE-T operation without auto-negotiation.

Broadcom recommends enabling and using auto-negotiation. For systems that only need 1000BASE-T functionality, Broadcom recommends enabling auto-negotiation with only 1000BASE-T being advertised, and that the advertising bits for all other modes be disabled.

</td></tr>
<tr><td>F1000</td><td>I_{PU}, SOR, D</td></tr>
<tr><td>FDX</td><td>I_{PU}, SOR, D</td><td>

Full-duplex Select. Active high**.** This input sets the default value of the Copper Interface Manual Duplex Mode bit in the MII Control register 0x00, bit[8]. This ball also sets the default value of the auto-negotiation advertised abilities for 10BASE-T, 100BASE-TX, and 1000BASE-T full-duplex capability. After reset, all duplex mode bits are under software control.

</td></tr>
<tr><td>HUB</td><td>I_{PD}, SOR, D</td><td>

Repeater Select. Active high. This input sets the default value of both the Hub/DTE bit and the master/slave configuration value bit in 1000BASE-T Control register (0x09, bits[11:10]) After reset, both bits are under software control.

</td></tr>
</table>

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
Misc. Configuration		
Note: V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 9: "Timing and AC Characteristics".		
SUPER_I	$I_{PD, CS, D}$	**Super Isolate Select.** Active high. Enable Super Isolate mode for all copper ports when pulled high. See "Super Isolate Mode" on page 47.
CRNT_LED_EN	$I_{PU, CS, D}$	**Constant Current LED Mode Enable.** Active high. Enables Constant Current LED Mode for all LED balls when pulled high or left floating. See "Constant Current LED Mode" on page 68 for details.
INTERRUPT	O_{PU}	**Interrupt.** Active low output with an internal pull-up resistor. Indicates the combined interrupt status of all ports.

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
LEDS These LEDs can be used when in: • Parallel LED mode • External Serial LED mode • Constant Current LED mode • Legacy Constant Current LED mode ***Note:*** V_{OH}, V_{OL}, V_{IH} and V_{IL} values are set by OVDD. See Section 10: “Electrical Characteristics”.		
LED0[0] LED0[1] LED0[2] LED0[3]/SD	I/O	**Port 0 LEDs.** Active-low. ***Note:*** **LED0[3]** can be programmed to INTERRUPT mode. See “Interrupt Interface” on page 59. ***Note:*** LED0[3]/SD can be used a signal detect/RX_LOS input.
LED1[0] LED1[1] LED1[2] LED1[3]/SD	I/O	**Port 1 LEDs.** Active-low. ***Note:*** LED1[3]/SD can be used a signal detect/RX_LOS input.
LED2[0] LED2[1] LED2[2] LED2[3]/SD	I/O	**Port 2 LEDs.** Active-low. ***Note:*** LED2[3]/SD can be used a signal detect/RX_LOS input.
LED3[0] LED3[1] LED3[2] LED3[3]/SD	I/O_T	**Port 3 Status LEDs.** ***Note:*** Active-low. ***Note:*** LED3[3]/SD can be used a signal detect/RX_LOS input.
LED4[0] LED4[1] LED4[2] LED4[3]/SD	I/O	**Port 4 LEDs.** ***Note:*** Active-low. ***Note:*** LED4[3]/SD can be used a signal detect/RX_LOS input.
LED5[0] LED5[1] LED5[2] LED5[3]/SD	II/O	**Port 5 LEDs.** ***Note:*** Active-low. ***Note:*** LED5[3]/SD can be used a signal detect/RX_LOS input.
LED6[0] LED6[1] LED6[2] LED6[3]/SD	I/O	**Port 6 LEDs.** Active-low. ***Note:*** LED6[3]/SD can be used a signal detect/RX_LOS input.
LED7[0] LED7[1] LED7[2] LED7[3]/SD	I/O	**Port 7 LEDs.** Active-low. ***Note:*** LED4[3]/SD can be used a signal detect/RX_LOS input.

Table 22: Ball Descriptions (Cont.)

Label	***I/O***	***Description***
External Serial LED Configuration		
SLED_EN	$I_{PD, CS, D}$	**External Serial LED Enable.** Active-high. Enables External Serial LED mode when pulled high.
SLED_IN	$I_{PD, CS, D}$	**External Serial LED Input.** Active-high. This input is used to clock in the external serial data.
SLED_STRB	$I_{PD,SOR, D}$	**External Serial LED Strobe.** Active-high. This input latches in 8-bits, 16-bits, 24-bits, or 32-bits from the SLED_IN serial data stream.
SLED_CLK	$I_{PD, CS}$	**External Serial LED Clock.** Active-high. This input clock strobes the SLED_IN serial stream data. Clock frequencies of up to 25 MHz are supported.
SLED_OUT	O_T	**External Serial LED Out.** Active High. Serial LED output that is used to daisy chain to the next BCM54285's SLED_IN.
Bias		
RDAC[1] RDAC[2]	O_A	**DAC Bias Resistor.** Adjusts the reference current of the transmitter digital-to-analog converter. A resistor of 6.04 kΩ ±1% is connected between the RDAC and GND.
Thermal Diode		
TDIODEP TDIODEN	I_A	**Thermal Diode.** Used to measure the junction temperature.
1.0V Power Supplies		
AVDDL	PWR	**Analog Low Voltage.** 1.0V.
DVDD	PWR	**Digital Low Core Voltage.** 1.0V.
PLLVDD	PWR	**PLL Voltage.** 1.0V.
SAVDDR	PWR	**SerDes Receiver Voltage.** 1.0V.
SAVDDT	PWR	**SerDes Transmitter Voltage.** 1.0V.
SPLLVDD	PWR	**SerDes PLL Voltage.** 1.0V.
QPVDD	PWR	**QSGMII PLL Voltage.** 1.0V
QRVDD	PWR	**QSGMII Receiver Voltage.** 1.0V
QTVDD	PWR	**QSGMII Transmitter Voltage.** 1.0V

Table 22: Ball Descriptions (Cont.)

Label	*I/O*	*Description*
3.3V Power Supplies		
AVDDH	PWR	**Analog High Voltage.** 3.3V.
BIASVDD	PWR	**Bias Voltage.** 3.3V.
CLKVDD	PWR	**Reference Clock Voltage.** 3.3V. The voltage supported for REFCLKP/ REFCLKN interface.
OVDD	PWR	**Digital Periphery Voltage.** 3.3V. This supply sets the V_{OH}, V_{OL}, V_{IH} and V_{IL} values for the following ball types: • Mode select • Strap options • Twisted-pair configuration • Test • LEDs • IEEE 1588 I/Os • Sync_E I/Os
Multivoltage Power Supplies		
OVDDMDIO	PWR	**MDIO/MDC Supply Voltage.** 1.2V, 2.5V, or 3.3V. The voltage supported by the MDIO interface is set by MDIO_LVL and applying 1.2V, 2.5V or 3.3V to OVDDMDIO. • MDIO_LVL = 0: OVDDMDIO operates at 1.2V. • MDIO_LVL = 1: OVDDMDIO can operate at either 2.5V or 3.3V.
OVDDJTAG	PWR	**JTAG Pad Supply Voltage.** 1.8V, 2.5V or 3.3V. The voltage supported by the JTAG interface is set by applying 1.8V, 2.5V or 3.3V to the OVDDJTAG.
Grounds		
GND	GND	**Ground.** 0.0V. Ground. All ground balls must be connected to a ground plane through vias.
No Connects		
DNC	DNC	**Do Not Connect.** These balls are for test purposes only and should not be connected. These balls must be left floating. Do not connect these balls together.
NC	NC	**Not Connected.** These balls have no internal connection in the device package. Leave these balls floating.

Section 6: Ball Locations

Ball Location Diagram

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A																				
B																				
C																				
D																				
E																				
F																				
G																				
H																				
J																				
K																				
L																				
M																				
N																				
P																				
R																				
T																				
U																				
V																				
W																				
Y																				

Figure 14: Top View: Ball Location Diagram

Ball Name Location Diagrams

	1	2	3	4	5	6	7	8	9	10
A	TDN_4_0	TDP_4_0	GND	$\overline{\text{LED7[1]}}$	$\overline{\text{LED4[1]}}$	$\overline{\text{LED6[1]}}$	SDA	DNC	$\overline{\text{LED4[2]}}$	MDIO[1]
B	TDP_4_1	TDN_4_1	GND	$\overline{\text{RESET}}$	$\overline{\text{LED4[3]}}$	$\overline{\text{LED6[0]}}$	OVDD	SCL	SYNC_IN	MDC[2]
C	TDN_4_2	TDP_4_2	GND	TEST[0]	$\overline{\text{LED4[0]}}$	$\overline{\text{LED5[0]}}$	GND	GND	OVDDMDIO	$\overline{\text{LED5[2]}}$
D	TDP_4_3	TDN_4_3	GND	SCLK_FREQ[0]	SCLK_TYPE[1]	$\overline{\text{LED5[3]}}$	$\overline{\text{LED5[1]}}$	FDX	BSC_EN[1]	DNC
E	TDN_5_3	TDP_5_3	GND	$\overline{\text{TRST}}$	OVDDJTAG	SCLK_TYPE[0]	$\overline{\text{LED7[2]}}$	ANEN	MDC[1]	SUPER_I
F	TDP_5_2	TDN_5_2	GND	GND	GND	TDO	$\overline{\text{LED7[0]}}$	MDIO_SEL[1]	MDIO[2]	PHYA_REV
G	TDN_5_1	TDP_5_1	GND	GND	GND	GND	DVDD	SYNC_OUT	$\overline{\text{LED3[2]}}$	SLED_CLK
H	TDP_5_0	TDN_5_0	GND	AVDDL	AVDDH	GND	GND	GND	GND	GND
J	TDN_6_0	TDP_6_0	GND	RDAC[2]	AVDDH	GND	GND	GND	GND	GND
K	TDP_6_1	TDN_6_1	GND	BIASVDD[2]	PLLVDD[2]	AVDDH	GND	GND	GND	GND
L	TDN_6_2	TDP_6_2	GND	TVCO[2]	AVDDL	AVDDL	GND	GND	GND	GND
M	TDP_6_3	TDN_6_3	GND	GND	GND	AVDDH	GND	GND	GND	GND
N	TDN_7_3	TDP_7_3	GND	DVDD	DVDD	GND	GND	GND	GND	GND
P	TDP_7_2	TDN_7_2	GND	SLED_OUT	REC_CLK[2]	OVDD	GND	SAVDDR	SAVDDT	SPLLVDD
R	TDN_7_1	TDP_7_1	GND	MDIO_SEL[2]	REC_CLK[1]	PHYA[4]	PHYA[2]	$\overline{\text{LED7[3]}}$	$\overline{\text{LED0[3]}}$	GND
T	TDP_7_0	TDN_7_0	GND	CRNT_LED_EN	GND	PHYA[3]	PHYA[1]	$\overline{\text{LED6[3]}}$	$\overline{\text{INTERRUPT}}$	GND
U	GND	GND	GND	GND	DNC	GND	PHYA[0]	$\overline{\text{LED1[3]}}$	GND	SRXDN[3]
V	GND	GND	GND	GND	DNC	GND	GND	GND	GND	SRXDP[3]
W	STXDP[7]	SRXDP[7]	STXDP[6]	SRXDP[6]	STXDP[5]	SRXDP[5]	STXDP[4]	SRXDP[4]	STXDP[3]	STXDP[2]
Y	STXDN[7]	SRXDN[7]	STXDN[6]	SRXDN[6]	STXDN[5]	SRXDN[5]	STXDN[4]	SRXDN[4]	STXDN[3]	STXDN[2]

Figure 15: Top View: Ball Name Location Diagram (Figure 1 of 2)

11	12	13	14	15	16	17	18	19	20	
BSC_EN[0]	DNC	DNC	$\overline{\text{LED6[2]}}$	LED2[1]	INTF_SEL[1]	SPD0	GND	TDP_3_0	TDN_3_0	**A**
DNC	SLED_STRB	TEST[1]	DNC	$\overline{\text{LED1[2]}}$	DNC	DNC	GND	TDN_3_1	TDP_3_1	**B**
TDIODEP	HUB	SLED_IN	OVDD	$\overline{\text{LED1[1]}}$	TMS	$\overline{\text{LED3[1]}}$	GND	TDP_3_2	TDN_3_2	**C**
TDIODEN	TDI	$\overline{\text{LED2[0]}}$	GND	TCK	$\overline{\text{LED0[1]}}$	DNC	GND	TDN_3_3	TDP_3_3	**D**
DNC	F1000	$\overline{\text{LED3[3]}}$	LOC_REC_CLK[1]	DVDD	$\overline{\text{LED2[3]}}$	$\overline{\text{LED3[0]}}$	GND	TDP_2_3	TDN_2_3	**E**
DNC	INTF_SEL[0]	SLED_EN	GND	DVDD	$\overline{\text{LED0[0]}}$	LOC_REC_CLK[2]	GND	TDN_2_2	TDP_2_2	**F**
$\overline{\text{LED0[2]}}$	$\overline{\text{LED1[0]}}$	$\overline{\text{LED2[2]}}$	GND	OVDDJTAG	GND	AVDDL	GND	TDP_2_1	TDN_2_1	**G**
GND	GND	GND	GND	GND	GND	GND	GND	TDN_2_0	TDP_2_0	**H**
GND	GND	GND	GND	AVDDH	TVCO[1]	BIASVDD[1]	GND	TDP_1_0	TDN_1_0	**J**
GND	GND	GND	GND	PLLVDD[1]	AVDDH	RDAC[1]	GND	TDN_1_1	TDP_1_1	**K**
GND	GND	GND	GND	AVDDL	AVDDH	AVDDH	GND	TDP_1_2	TDN_1_2	**L**
GND	GND	GND	GND	MDIO_LVL	AVDDL	GND	GND	TDN_1_3	TDP_1_3	**M**
GND	GND	GND	GND	DVDD	DNC	OVDD	GND	TDP_0_3	TDN_0_3	**N**
SPLLVDD	DNC	SAVDDT	GND	DNC	DNC	SCLK_FREQ[1]	GND	TDN_0_2	TDP_0_2	**P**
DNC	DNC	SAVDDR	GND	DNC	CLKVDD	GND	GND	TDP_0_1	TDN_0_1	**R**
DNC	SRXDN[0]	SRXDP[0]	GND	DNC	REFCLKN	REFCLKP	GND	TDN_0_0	TDP_0_0	**T**
SRXDP[2]	SRXDP[1]	STXDP[0]	STXDN[0]	GND	GND	GND	GND	GND	GND	**U**
SRXDN[2]	SRXDN[1]	QRVDD	QRVDD	GND	QPVDD	QPVDD	QRVDD	GND	GND	**V**
STXDP[1]	GND	RDN0	GND	TDN0	DNC	DNC	RDN1	QTVDD	TDP1	**W**
STXDN[1]	GND	RDP0	QTVDD	TDP0	DNC	DNC	RDP1	QTVDD	TDN1	**Y**

Figure 16: Top View: Ball Name Location Diagram (Figure 2 of 2)

Section 7: Ball Assignments

Ballout Listed by Ball Number

Table 23: Ballout Listed by Ball Number

Ball	Ball Name
A01	TDN_4_0
A02	TDP_4_0
A03	GND
A04	LED7[1]
A05	LED4[1]
A06	LED6[1]
A07	SDA
A08	DNC
A09	LED4[2]
A10	MDIO[1]
A11	BSC_EN[0]
A12	DNC
A13	DNC
A14	LED6[2]
A15	LED2[1]
A16	INTF_SEL[1]
A17	SPD0
A18	GND
A19	TDP_3_0
A20	TDN_3_0
B01	TDP_4_1
B02	TDN_4_1
B03	GND
B04	RESET
B05	LED4[3]
B06	LED6[0]
B07	OVDD
B08	SCL
B09	SYNC_IN
B10	MDC[2]
B11	DNC
B12	SLED_STRB
B13	TEST[1]
B14	DNC
B15	LED1[2]
B16	DNC
B17	DNC
B18	GND
B19	TDN_3_1
B20	TDP_3_1
C01	TDN_4_2
C02	TDP_4_2
C03	GND
C04	TEST[0]
C05	LED4[0]
C06	LED5[0]
C07	GND
C08	GND
C09	OVDDMDIO
C10	LED5[2]
C11	TDIODEP
C12	HUB
C13	SLED_IN
C14	OVDD
C15	LED1[1]
C16	TMS
C17	LED3[1]
C18	GND
C19	TDP_3_2
C20	TDN_3_2
D01	TDP_4_3
D02	TDN_4_3
D03	GND
D04	SCLK_FREQ[0]
D05	SCLK_TYPE[1]
D06	LED5[3]
D07	LED5[1]
D08	FDX
D09	BSC_EN[1]
D10	DNC
D11	TDIODEN
D12	TDI
D13	LED2[0]
D14	GND
D15	TCK
D16	LED0[1]
D17	DNC
D18	GND
D19	TDN_3_3
D20	TDP_3_3
E01	TDN_5_3
E02	TDP_5_3
E03	GND
E04	TRST
E05	OVDDJTAG
E06	SCLK_TYPE[0]
E07	LED7[2]
E08	ANEN
E09	MDC[1]
E10	SUPER_I
E11	DNC
E12	F1000
E13	LED3[3]
E14	LOC_REC_CLK[1]
E15	DVDD
E16	LED2[3]
E17	LED3[0]
E18	GND
E19	TDP_2_3
E20	TDN_2_3
F01	TDP_5_2
F02	TDN_5_2
F03	GND
F04	GND
F05	GND
F06	TDO
F07	LED7[0]
F08	MDIO_SEL[1]
F09	MDIO[2]
F10	PHYA_REV
F11	DNC
F12	INTF_SEL[0]
F13	SLED_EN
F14	GND
F15	DVDD
F16	LED0[0]
F17	LOC_REC_CLK[2]
F18	GND
F19	TDN_2_2
F20	TDP_2_2
G01	TDN_5_1
G02	TDP_5_1
G03	GND
G04	GND
G05	GND
G06	GND
G07	DVDD
G08	SYNC_OUT
G09	LED3[2]
G10	SLED_CLK
G11	LED0[2]
G12	LED1[0]
G13	LED2[2]
G14	GND
G15	OVDDJTAG
G16	GND
G17	AVDDL
G18	GND
G19	TDP_2_1
G20	TDN_2_1

Ball	Ball Name
H01	TDP_5_0
H02	TDN_5_0
H03	GND
H04	AVDDL
H05	AVDDH
H06	GND
H07	GND
H08	GND
H09	GND
H10	GND
H11	GND
H12	GND
H13	GND
H14	GND
H15	GND
H16	GND
H17	GND
H18	GND
H19	TDN_2_0
H20	TDP_2_0
J01	TDN_6_0
J02	TDP_6_0
J03	GND
J04	RDAC[2]
J05	AVDDH
J06	GND
J07	GND
J08	GND
J09	GND
J10	GND
J11	GND
J12	GND
J13	GND
J14	GND
J15	AVDDH
J16	TVCO[1]
J17	BIASVDD[1]
J18	GND
J19	TDP_1_0
J20	TDN_1_0
K01	TDP_6_1
K02	TDN_6_1
K03	GND
K04	BIASVDD[2]
K05	PLLVDD[2]
K06	AVDDH
K07	GND
K08	GND
K09	GND
K10	GND
K11	GND
K12	GND
K13	GND
K14	GND
K15	PLLVDD[1]
K16	AVDDH
K17	RDAC[1]
K18	GND
K19	TDN_1_1
K20	TDP_1_1
L01	TDN_6_2
L02	TDP_6_2
L03	GND
L04	TVCO[2]
L05	AVDDL
L06	AVDDL
L07	GND
L08	GND
L09	GND
L10	GND
L11	GND
L12	GND
L13	GND
L14	GND
L15	AVDDL
L16	AVDDH
L17	AVDDH
L18	GND
L19	TDP_1_2
L20	TDN_1_2
M01	TDP_6_3
M02	TDN_6_3
M03	GND
M04	GND
M05	GND
M06	AVDDH
M07	GND
M08	GND
M09	GND
M10	GND
M11	GND
M12	GND
M13	GND
M14	GND
M15	MDIO_LVL
M16	AVDDL
M17	GND
M18	GND
M19	TDN_1_3
M20	TDP_1_3
N01	TDN_7_3
N02	TDP_7_3
N03	GND
N04	DVDD
N05	DVDD
N06	GND
N07	GND
N08	GND
N09	GND
N10	GND
N11	GND
N12	GND
N13	GND
N14	GND
N15	DVDD
N16	DNC
N17	OVDD
N18	GND
N19	TDP_0_3
N20	TDN_0_3
P01	TDP_7_2
P02	TDN_7_2
P03	GND
P04	SLED_OUT
P05	REC_CLK[2]
P06	OVDD
P07	GND
P08	SAVDDR
P09	SAVDDT
P10	SPLLVDD
P11	SPLLVDD
P12	DNC
P13	SAVDDT
P14	GND
P15	DNC
P16	DNC
P17	SCLK_FREQ[1]
P18	GND
P19	TDN_0_2
P20	TDP_0_2
R01	TDN_7_1
R02	TDP_7_1
R03	GND
R04	MDIO_SEL[2]
R05	REC_CLK[1]
R06	PHYA[4]
R07	PHYA[2]
R08	$\overline{\text{LED7[3]}}$
R09	$\overline{\text{LED0[3]}}$
R10	GND
R11	DNC
R12	DNC
R13	SAVDDR
R14	GND
R15	DNC
R16	CLKVDD
R17	GND
R18	GND
R19	TDP_0_1
R20	TDN_0_1
T01	TDP_7_0
T02	TDN_7_0
T03	GND
T04	CRNT_LED_EN
T05	GND
T06	PHYA[3]
T07	PHYA[1]
T08	$\overline{\text{LED6[3]}}$
T09	$\overline{\text{INTERRUPT}}$
T10	GND
T11	DNC
T12	SRXDN[0]

Ball	Ball Name
T13	SRXDP[0]
T14	GND
T15	DNC
T16	REFCLKN
T17	REFCLKP
T18	GND
T19	TDN_0_0
T20	TDP_0_0
U01	GND
U02	GND
U03	GND
U04	GND
U05	DNC
U06	GND
U07	PHYA[0]
U08	LED1[3]
U09	GND
U10	SRXDN[3]
U11	SRXDP[2]
U12	SRXDP[1]
U13	STXDP[0]
U14	STXDN[0]
U15	GND
U16	GND
U17	GND
U18	GND
U19	GND
U20	GND
V01	GND
V02	GND
V03	GND
V04	GND
V05	DNC
V06	GND
V07	GND
V08	GND
V09	GND
V10	SRXDP[3]
V11	SRXDN[2]
V12	SRXDN[1]
V13	QRVDD
V14	QRVDD
V15	GND
V16	QPVDD
V17	QPVDD
V18	QRVDD
V19	GND
V20	GND
W01	STXDP[7]
W02	SRXDP[7]
W03	STXDP[6]
W04	SRXDP[6]
W05	STXDP[5]
W06	SRXDP[5]
W07	STXDP[4]
W08	SRXDP[4]
W09	STXDP[3]
W10	STXDP[2]
W11	STXDP[1]
W12	GND
W13	RDN0
W14	GND
W15	TDN0
W16	DNC
W17	DNC
W18	RDN1
W19	QTVDD
W20	TDP1
Y01	STXDN[7]
Y02	SRXDN[7]
Y03	STXDN[6]
Y04	SRXDN[6]
Y05	STXDN[5]
Y06	SRXDN[5]
Y07	STXDN[4]
Y08	SRXDN[4]
Y09	STXDN[3]
Y10	STXDN[2]
Y11	STXDN[1]
Y12	GND
Y13	RDP0
Y14	QTVDD
Y15	TDP0
Y16	DNC
Y17	DNC
Y18	RDP1
Y19	QTVDD
Y20	TDN1

Section 8: Operational Description

Power Sequencing

There are no specific power sequencing requirements for the BCM54285. It is recommended that the higher supply be brought up first (for example, 3.3V first and then 1.0V) or at the same time as the lower supplies. This is only a recommendation and not a requirement.

Resetting the BCM54285

Hardware Reset

$\overline{\text{RESET}}$ resets all internal nodes to a known state. Hardware reset is accomplished by holding $\overline{\text{RESET}}$ low for at least 10 ms after the power supply voltages and clocks are stable. Once RESET is brought high, the PHY completes its reset sequence within 20 µs. All outputs are inactive until the PHY has completed its reset sequence. The PHY keeps the inputs inactive for 5 ms after the deassertion of hardware reset. The hardware configuration balls and the PHY address balls are read on the deassertion of hardware reset. See "Reset Timing" on page 107.

Caution! $\overline{\text{RESET}}$ must be asserted during power-up.

Software Reset

The BCM54285 has multiple reset bits to reset the different blocks/registers are listed below and in Table 24: "Reset Summary," on page 92.

- Per Port Register Reset
- Global RDB Register Reset
- Clause 45 Register Reset
- QSGMII Register Reset

Per Port Register Reset

The BCM54285 has two methods for performing per port register software reset.

The first method behaves like a hardware reset except it is done on a port-by-port basis. This reset will set the PHY registers listed below to the default values and hardware strap balls that are labelled *sample on reset* (SOR) are relatched.

- IEEE Registers (0x00 to 0x0F)

- Per port RDB registers (RDB_Registers, offset 0x00 to offset 0x2FF)

To enable software reset set register 0x00, bit[15] = 1’b1. This needs to be done to every port that needs to be reset. This bit is self-clearing.

The second method issues a soft-reset for 640 ns that will clear the status registers/bits listed below and leave MDIO control registers/bits alone. Strap options are not relatched.

- IEEE registers (0x00 to 0x0F)
- Per port RDB registers (RDB_Registers, offset 0x00 to offset 0x2FF)

This reset is done on a port-by-port basis. To enable the software reset, set RDB_Reg, offset 0x021, bit[0] = 1’b0 to access the copper register space and then set RDB_Register, offset 0x070, bit[0] = 1’b1. This needs to be done to every port that needs to be reset. This bit is self-clearing.

Note: This reset does not affect the QSGMII Registers.

Global RDB Register Reset

Global RDB register reset will reset the global RDB registers (RDB_Registers, offset 0x800 to offset 0xAFF) to their default values. This reset is done on a PHY basis. To enable the software reset, set RDB_Register, offset 0x82B, bit[15] = 1’b1. This needs to be done to PORT 0’s PHY address. This bit is self-clearing.

Note: This reset does not affect the following registers:

- Per port register (Reg. 0x00 to 0x0F and RDB_Registers, offset 0x00 to offset 0x2F)
- Clause 45 registers
- QSGMII registers

Clause 45 Register Reset

Clause 45 register reset will set the EEE block and registers to their default values. To enable the reset, set Clause 45 DEVAD 0x1, Address 0x0, bit[15] = 1’b0. This needs to be done to PORT 0’s PHY address. This bit is self-clearing.

Note: his reset does not affect the following registers:

- Per port register (Reg. 0x00 to 0x0F and RDB_Registers, offset 0x00 to offset 0x2F)
- Global RDB registers (RDB_Registers, offset 0x800 to offset 0xAFF)
- QSGMII registers

QSGMII Register Reset

QSGMII register reset will set the QSGMII block and registers to their default values. There are two resets bits, one for QSGMII Quad 0 and the other for QSGMII Quad 1.

To enable QSGMII Quad 0 reset, set QSGMII register, BAR = 0x0000, REGAD = 0x00, AER = 0x000. To enable QSGMII Quad 0 reset, set QSGMII register, BAR = 0x0000, REGAD = 0x00, AER = 0x004. This needs to be done to PORT 0’s PHY address. These bits are self-clearing.

Table 24: Reset Summary

Reset	***Blocks Affected***	***Registers Affected***
Per PHY hardware reset (method 1)	• Twisted-pair • Fiber • Reference clock • LEDs • Cable diagnostics • Hardware strap balls • IEEE 1588 • Voltage/temp monitor • Sync_E • EEE • AutogrEEEn	• RDB_Reg, offset 0x021, bit[0] = 1’b0 – Reg. 0x0 to 0xF • RDB_Reg, offset 0x021, bit[0] = 1’b1 – Reg. 0x0 to 0xF • RDB_Reg., offset 0x00 to offset 0x2FF • RDB_Reg, offset 0x800 to offset 0xAFF • Clause 45 DEVAD 0x1 to DEVAD 0x7 • QSGMII Reg. BAR 0x0 to BAR 0x8300
Per port register reset (method 2) • RDB_Reg, offset 0x021, bit[0] = 1’b0 – Reg. 0x00, bit[15] = 1’b1	• Twisted-pair • Fiber • Reference clock • LEDs • Cable diagnostics • Hardware strap balls	• RDB_Reg, offset 0x021, bit[0] = 1’b0 – Reg. 0x0 to 0xF • RDB_Reg., offset 0x00 to offset 0x2FF
Per port register reset • RDB_Reg, offset 0x021, bit[0] = 1’b0 – RDB_Reg, offset 0x070, bit[0] = 1’b1	NA	Status registers/bits only • Reg. 0x0 to 0xF • RDB_Reg, offset 0x00 to offset 0x2FF
Global RDB register reset • RDB_Reg, offset 0x82B, bit[15] = 1’b1	• IEEE 1588 • Voltage/temp monitor • Sync_E	• RDB_Reg, offset 0x800 to offset 0xAFF
Clause 45 register reset • Clause 45 DEVAD 0x1, Address 0x0, bit[15] = 1’b0	• EEE	• Clause 45 DEVAD 0x1 to DEVAD 0x7
QSGMII register reset QSGMII[0] (QSGMII quad 0) • BAR = 0x0000, REGAD = 0x00, AER = 0x000 QSGMII[1] (QSGMII quad 1) • BAR = 0x0000, REGAD = 0x00, AER = 0x0004	• QSGMII	• QSGMII Reg. BAR 0x0 to BAR 0x8300

Reference Clock Using REFCLKP/N

The BCM54285 supports the various clock modes shown in Table 25.

Table 25: Reference Clock Modes

Mode	***SCLK_TYPE[1:0]***	***SCLK_FREQ[1:0]***
25 MHz crystal	10	00
25 MHz single-ended CMOS clock	01	00
25 MHz differential clock	00	00
125 MHz single-ended CMOS clock	01	01
125 MHz differential clock	00	01
156.25 MHz single-ended CMOS clock	01	11
156.25 MHz differential clock	00	11
312.5 MHz differential clock	00	10

CML Differential Mode

Configurations for the CML Differential Clock modes are shown in Table 26. Figure 17 on page 94 shows a typical differential clock connection. REFCLKP/REFCLKN have an internal 100 ohm differential must be AC coupled to the differential clock source with a 0.1µF cap.

Table 26: Differential Reference Clock Modes

Mode	***SCLK_TYPE[1:0]***	***SCLK_FREQ[1:0]***
25 MHz differential clock	00	00
125 MHz differential clock	00	01
156.25 MHz differential clock	00	11
312.5 MHz differential clock	00	10

.

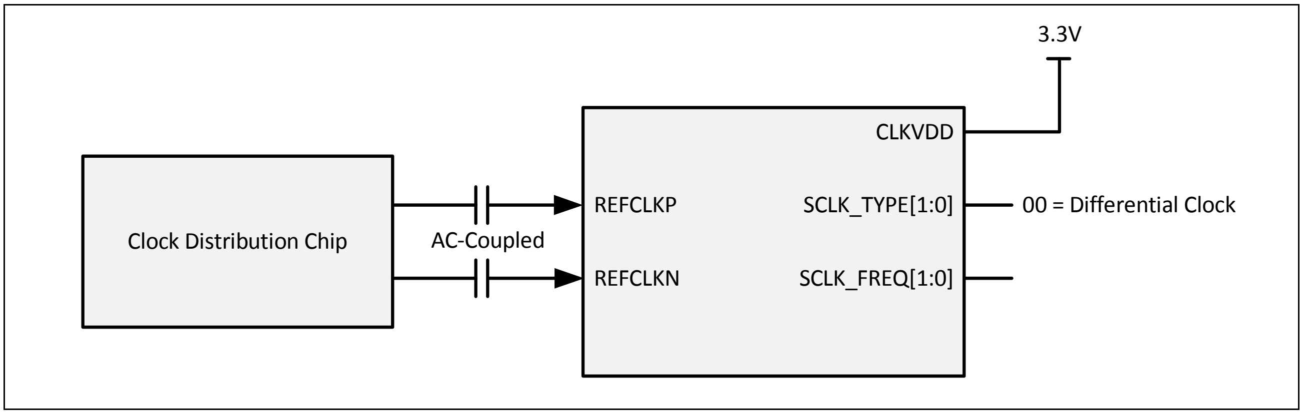


Figure 17: Differential Clock Input

Single-Ended CMOS Mode

Configurations for Single-Ended CMOS clock modes are shown in Table 27. Figure 18 shows a typical Single-Ended CMOS clock connection. The REFCLKN input must be connected directly to ground. R1 = $Z_{Trace} - Z_{Driver}$ where R1 is a series termination resistor used to match the output impedance of the Oscillator (Z_{Driver}) to the trace impedance (Z_{Trace}).

Table 27: Single-Ended CMOS Reference Clock Modes

Mode	*SCLK_TYPE[1:0]*	*SCLK_FREQ[1:0]*
25 MHz single-ended CMOS clock	01	00
125 MHz single-ended CMOS clock	01	01
156.25 MHz single-ended CMOS clock	01	11

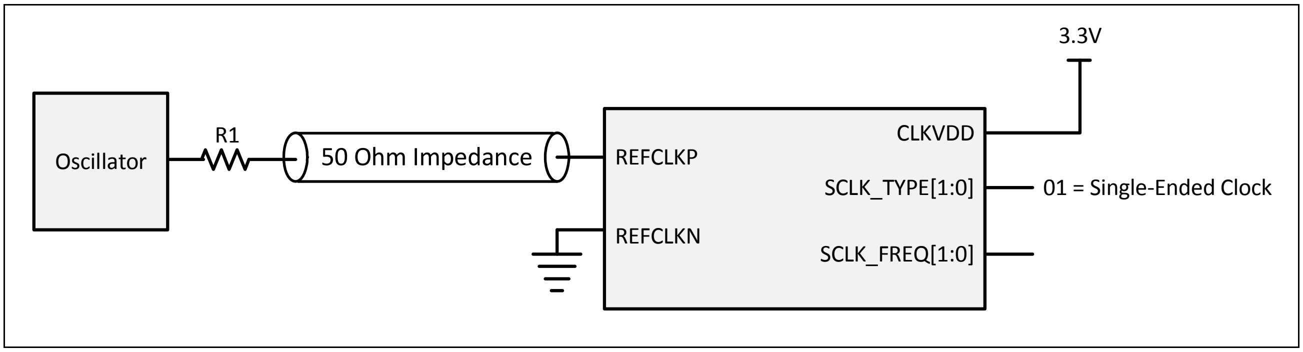


Figure 18: Single-Ended CMOS Input

Crystal Mode

Configurations for crystals are shown in Table 28. Figure 19 shows a typical crystal connection and Table 29 shows typical crystal parameters. The BCM54285 can only support a 25 MHz crystal.

Table 28: Crystal Clock Modes

Mode	*SCLK_TYPE[1:0]*	*SCLK_FREQ[1:0]*
25 MHz crystal	10	00

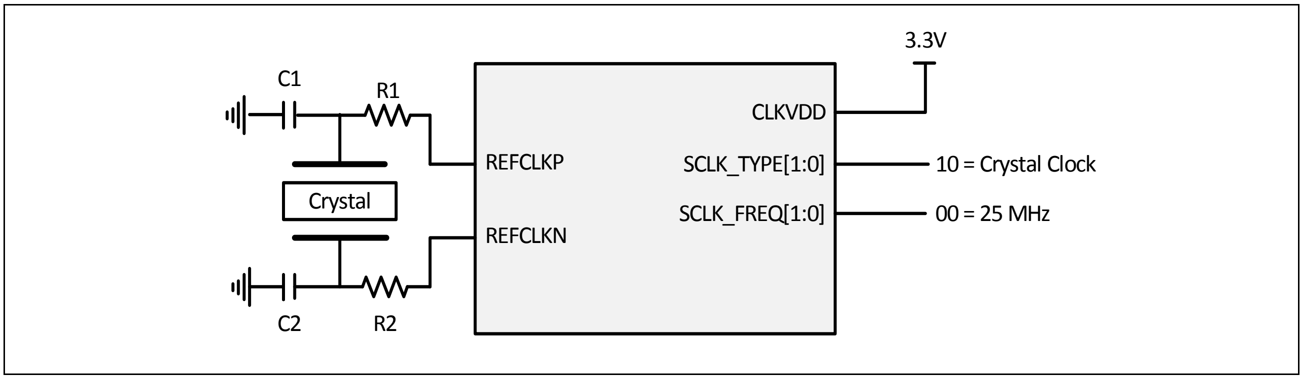


Figure 19: Crystal Application

Table 29: Crystal Clock Parameters

Parameter	*Value*
Frequency	25.000 MHz
Tolerance	±50 ppm
Operating mode	Parallel resonance
ESR	< 40Ω
Crystal loading capacitance	18 pF typical
Crystal aging	5 ppm per year maximum
Crystal stability	±30 ppm minimum
Crystal drive level	500 µW

The differential input is designed to operate properly with an external crystal that has a maximum ESR of less than 40Ω and is parallel-resonant with a standard 18 pF load. There is not a general value for the off-chip components that work for all cases. Their values will depend on the crystal, board layout, and operation voltage. The values in Table 30 on page 96 will get the oscillator working, but proper operation and characterization needs to be verified.

Table 30: Crystal Starting Values

C1	C2	R1	R2
27 pF to 30 pF	27 pF to 30 pF	10Ω	10Ω

Q Adjustment

There are two series resistors, R1 and R2, that have a typical value of 10Ω. R1 and R2 are used to adjust the Q of this circuit. No current-limiting series resistor to the crystal is needed because the current to the clock crystal is limited on the chip, thus eliminating the need for an additional external component. The loading capacitors for the remaining clock crystal tank circuit are determined based on the clock crystal data sheet loading specification.

Determining Load Capacitance (C_{load})

The load capacitance is primarily determined by the specified frequency. The following equation gives the approximate load capacitance, where C_{stray} is the stray capacitance (board parasitics, and so on).

$$C_{load} = C_{stray} + \frac{C_1 \times C_2}{C_1 + C_2}$$

C_{stray} is usually in the range of 2 pF to 5 pF. The crystal is specified with ±50 ppm tolerance for a given load capacitance. Therefore adjust the tank circuit loading capacitance values for C1 and C2 to force the oscillation frequency within the specified tolerance. If the load capacitance is too small, the frequency will be too high. If the board parasitic capacitances are too large or the crystal's specified load capacitance is too small, it may become difficult to adjust the frequency within limits. In this case, either choose another crystal with a larger specified load capacitance or reduce the parasitic capacitance.

Note:
- Usually $C_1 = C_2 = 2 \times$ Cload is used to adjust the frequency of the oscillator.
- R1 and R2 on REF_CLKP and REF_CLKN are used to adjust the Q of the circuit.

The following is an example calculation for a crystal with a typical load capacitance (C_{load} = 18 pF) and C_{stray} estimated at 4 pF maximum. From the equation:

$$C_{load} = C_{stray} + \frac{C_1 \times C_2}{C_1 + C_2}$$

Let $C_1 = C_2 = C$

$$C = 2(C_{load} - C_{stray})$$

$C = C_1 = C_2 = 2 \times (18\ \text{pF} - 4\ \text{pF}) = 28\ \text{pF}$. The load capacitance value of 27 pF is chosen because this is an easily procured standard capacitor value.

Copper Loopback With Loopback Plug

This allows packets to be sent to the PHY's QSGMII receive input interface, to the copper twisted-pair interface, to the magnetics and RJ-45 connector and back to the PHY's QSGMII transmit interface. This mode is enabled on a per-port basis. The red dashed line in Figure 20 on page 98 shows the loopback path for port 0. In 1000BASE-T, 100BASE-TX, and 10BASE-T modes, a loopback plug must be inserted into the RJ-45 connector. The jumper block should have the following RJ-45 pins connected together:

- Pin 1 connected to pin 3
- Pin 2 connected to pin 6
- Pin 4 connected to pin 7
- Pin 5 connected to pin 8

Note: Copper loopback with loopback plug is not guaranteed to work with EEE or AutogrEEEn enabled. To exit the copper loopback with loopback plug, Broadcom recommends a software or hardware reset.

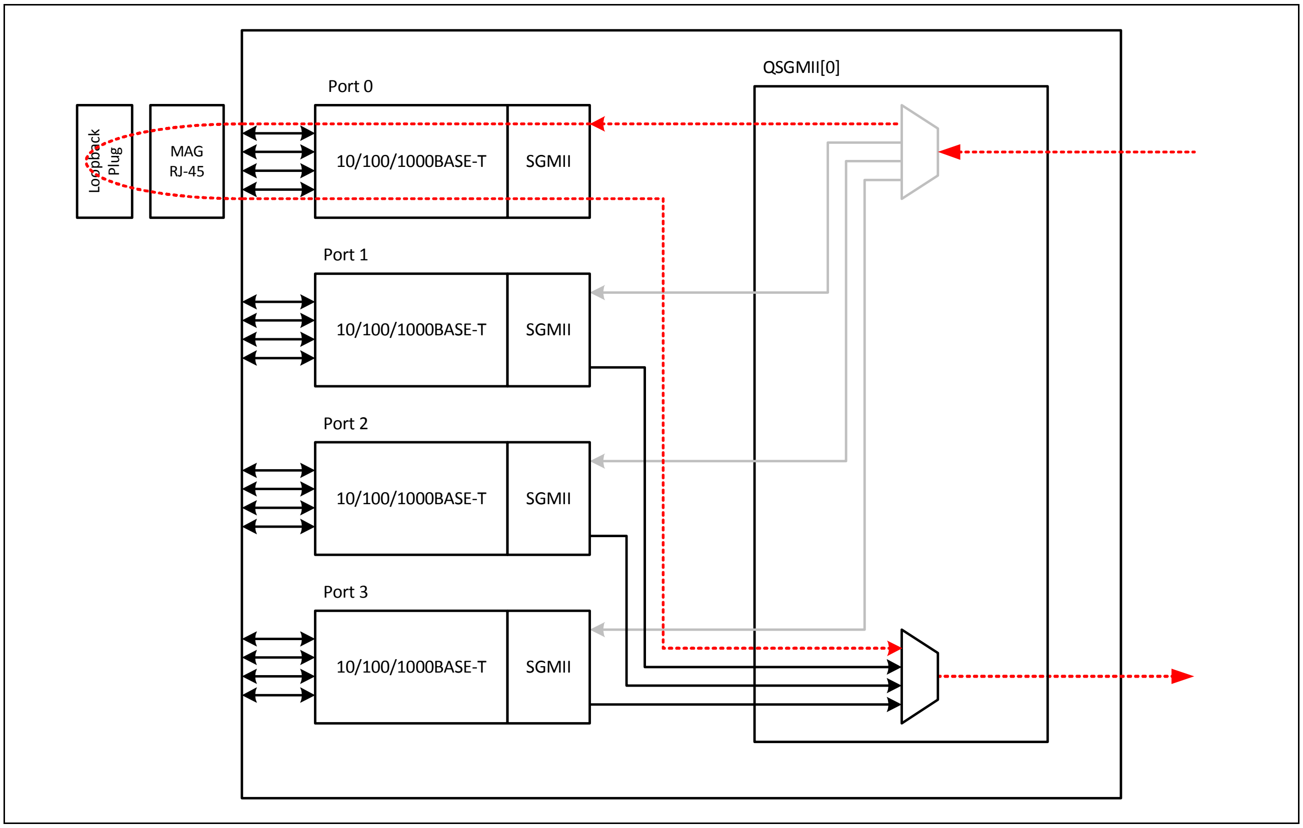


Figure 20: Copper Loopback Mode with Loopback Plug on Port 0

Table 31 through Table 33 on page 99 describe how the copper loopback with loopback plug is enabled for 1000BASE-T, 100BASE-TX, and 10BASE-T modes.

Table 31: 1000BASE-T Loopback With Loopback Plug

Register Writes	*Comments*
Write register 0x09 = 0x1800	Enable 1000BASE-T master mode.
Write register 0x00 = 0x0040	Enable force 1000BASE-T.
Write RDB_Register, offset 0x028 = 0x8400	Enable 1000BASE-T loopback mode with loopback plug.

Table 32: 100BASE-TX Loopback With Loopback Plug

Register Writes	*Comments*
Write register 0x00 = 0x2100	Enable force 100BASE-TX full-duplex mode.
Write RDB_Register, offset 0x028 = 0x8400	Enable 100BASE-TX Loopback mode with loopback plug.

Table 33: 10BASE-T Loopback With Loopback Plug

Register Writes	***Comments***
Write register 0x00 = 0x0100	Enable force 10BASE-T full-duplex mode.
Write RDB_Register, offset 0x028 = 0x8400	Enable 10BASE-T loopback mode with loopback plug.

Copper Loopback Without Loopback Plug

This allows packets to be sent to the PHY's QSGMII receive input interface, to the copper twisted-pair interface and back to the PHY's QSGMII transmit interface. This does not test the magnetics of RJ-45 connector. This is enabled on a per port basis. Figure 21 shows the loopback path for port 0.

Note: The copper loopback without the loopback plug does not work properly, if there is a jumper block or cable connected to the RJ-45 connector. The copper loopback without the loopback plug is not guaranteed to work with EEE or AutogrEEEn enabled. To exit copper loopback without the loopback plug, Broadcom recommends a software or hardware reset.

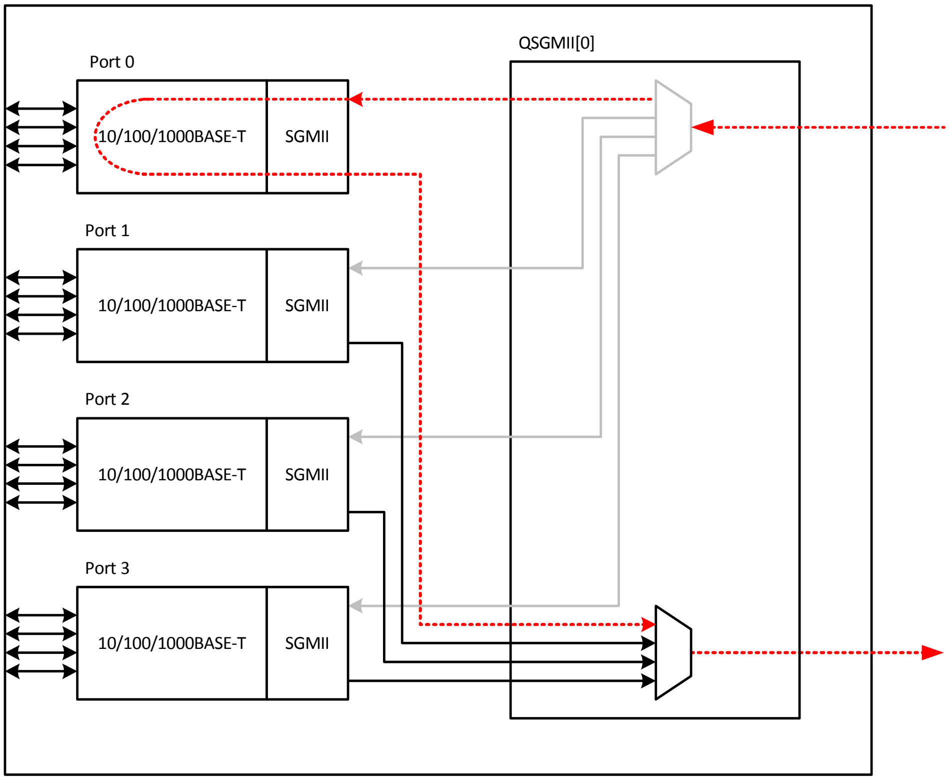


Figure 21: External Copper Loopback Mode without Loopback Plug on Port 0

Table 34 through Table 36 describe how the copper loopback mode is enabled for 1000BASE-T, 100BASE-TX, and 10BASE-T modes with and without a Loopback Plug.

Table 34: 1000BASE-T Loopback Without Loopback Plug

Register Writes	*Comments*
Write register 0x09 = 0x1800	Enable 1000BASE-T master mode.
Write register 0x00 = 0x0040	Enable force 1000BASE-T.
Write RDB_Register, offset 0x028 = 0x8400	Enable 1000BASE-T loopback mode.
Write RDB_Register, offset 0x02C = 0x4014	Enable 1000BASE-T loopback mode without loopback plug.

Table 35: 100BASE-TX Loopback Without Loopback Plug

Register Writes	*Comment*
Write register 0x00 = 0x2100	Enable force 100BASE-TX full-duplex mode.
Write RDB_Register, offset 0x028 = 0x8400	Enable 100BASE-TX loopback mode with loopback plug.
Write RDB_Register, offset 0x02C = 0x4014	Enable 100BASE-TX loopback mode without loopback plug.

Table 36: 10BASE-T Loopback Without Loopback Plug

Register Writes	*Comments*
Write register 0x00 = 0x0100	Enable force 10BASE-T full-duplex mode.
Write RDB_Register, offset 0x028 = 0x8400	Enable 10BASE-T loopback mode with loopback plug.
Write RDB_Register, offset 0x02C = 0x4014	Enable 10BASE-T loopback mode without loopback plug.

Copper Line-Side Loopback

This allows packets to be sent to the PHY's twisted-pair interface to the PHY's PCS layer and back to the PHY's twisted-pair interface. This mode is enabled on a per port basis. The red dashed line in Figure 22 on page 101 shows the loopback path for port 0.

Note: Copper line-side loopback is not guaranteed to work with EEE or AutogrEEEn enabled. To exit copper line-side loopback, Broadcom recommends a software or hardware reset.

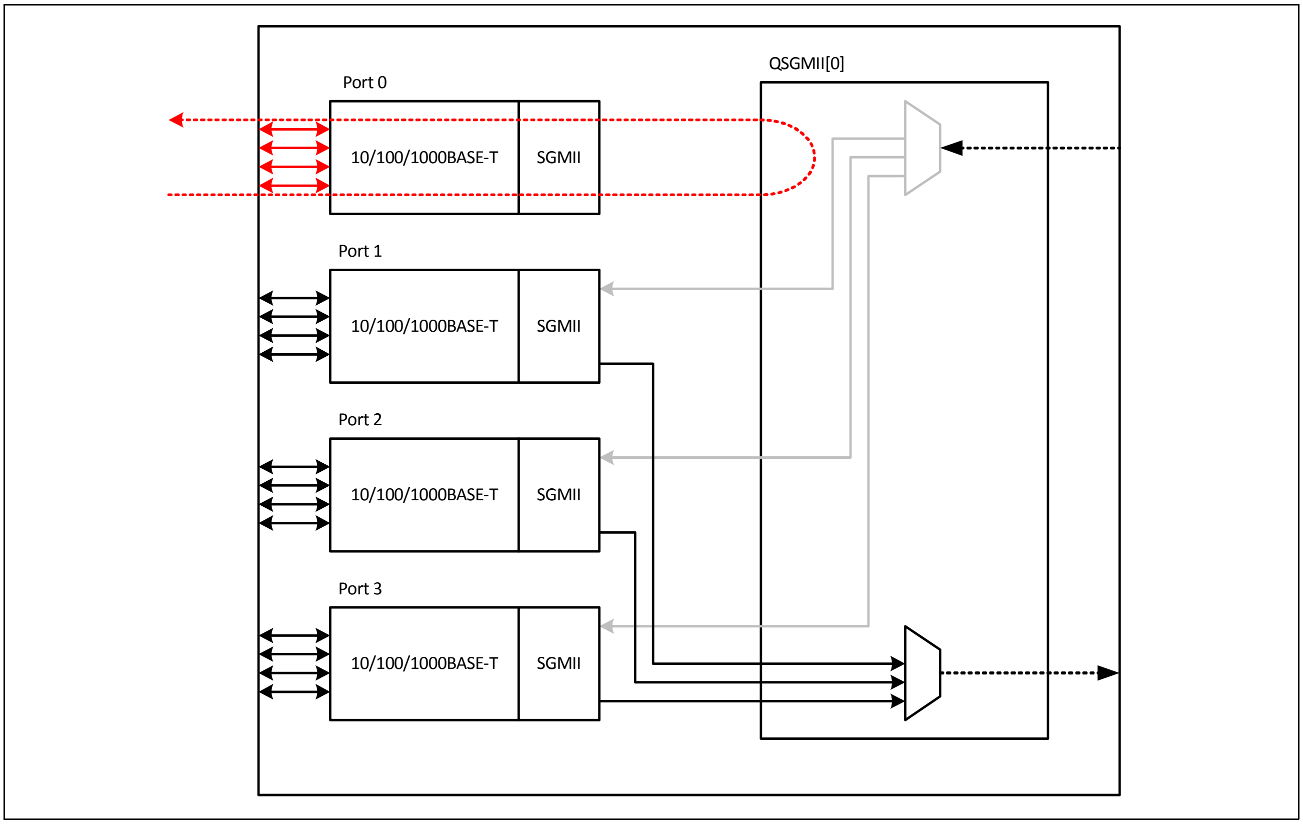


Figure 22: Copper Line-Side Loopback on Port 0

To enable copper line-side loopback, do the following writes:

- Write RDB_Register, offset 0x02C, bit[15] = 1'b1 (copper line-side loopback enable)
- Write register 0x0, bit[9] = 1'b1 (restart auto-negotiation)

1000BASE-X Line-Side Loopback

This allows packets to be sent to the PHY's 1000BASE-X interface to the PHY's PCS layer and back to the PHY's twisted-pair interface. This mode is enabled on a per port basis. The red dashed line in Figure 23 on page 102 shows the loopback path for port 0.

Note: To exit fiber line-side loopback, Broadcom recommends a software or hardware reset.

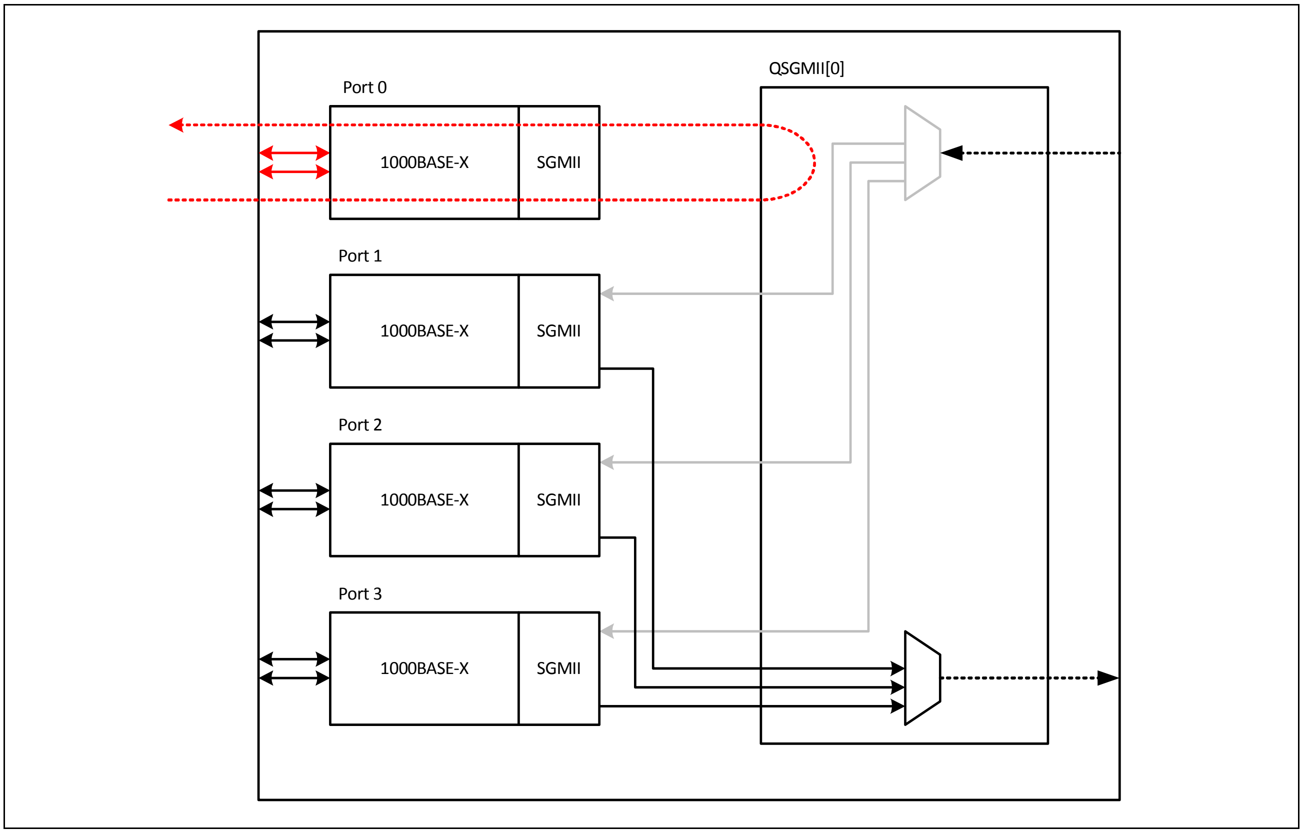


Figure 23: 1000BASE-X Line-Side Loopback on Port 0

To enable 1000BASE-X line-side loopback, do the following writes:

- Write RDB_Register, offset 0x23E = 0x7800
- Write RDB_Register, offset 0x021 = 0x7F06
- Write register 0x0 = 0x5140

Per Port QSGMII Packet Loopback

This allows data packets to be sent to the PHY’s QSGMII receive input to the PHY’s QSGMII PCS layer and back to the PHY’s QSGMII transmit output. This mode is enabled on a per port basis (all four ports per QSGMII can be enabled). The speed of the PHY port’s QSGMII lane must match the speed of the switch’s QSGMII lane. The red dashed line in Figure 24 on page 103 shows the loopback path for port 0.

Note: To exit per port QSGMII packet loopback, Broadcom recommends a software or hardware reset.

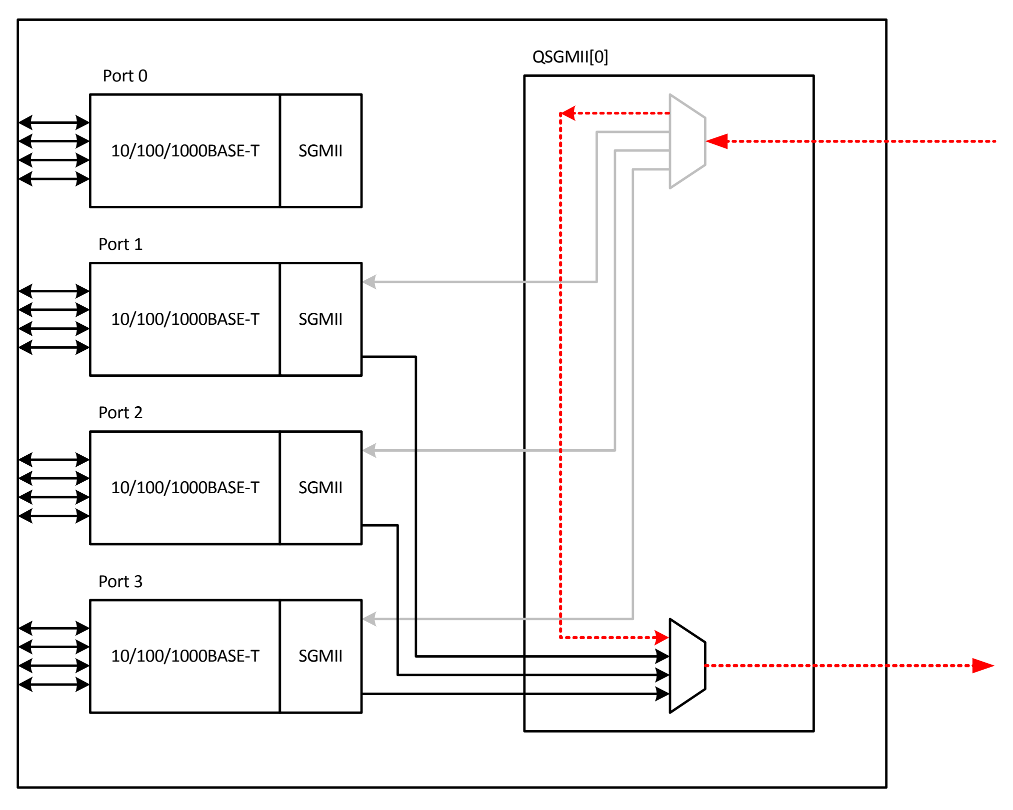


Figure 24: Per Port QSGMII Loopback on Port 0

To enable per port QSGMII loopback, do the following writes:

- Write register 0x1F = 0xFFD0 (set BAR to AER register)
  - Write register 0x1E = 0x0 (enable read/writes to port 0) and or
  - Write register 0x1E = 0x1 (enable read/writes to port 1) and or
  - Write register 0x1E = 0x2 (enable read/writes to port 2) and or
  - Write register 0x1E = 0x3 (enable read/writes to port 3) and or
  - Write register 0x1E = 0x4 (enable read/writes to port 4) and or
  - Write register 0x1E = 0x5 (enable read/writes to port 5) and or
  - Write register 0x1E = 0x6 (enable read/writes to Port 6) and or
  - Write register 0x1E = 0x7 (enable read/writes to Port 7) and or
  - Write register 0x1E = 0x1F (enable read/writes to all eight ports)
- Write QSGMII register (BAR = 0x8300, REGAD = 0x10) = 0x05A0 (enable loopback)

Per Port QSGMII/SGMII Packet Loopback

This allows data packets to be sent to the PHY's QSGMII receive input to the PHY's SGMII PCS layer and back to the PHY's QSGMII transmit output. This mode is enabled on a per port basis (All four ports per QSGMII can be enabled). The speed of the PHY Port's QSGMII lane must match the speed of the switch's QSGMII lane. The red dashed line in Figure 25 shows the loopback path for port 0.

Note: To exit per port QSGMII/SGMII packet loopback, Broadcom recommends a software or hardware reset.

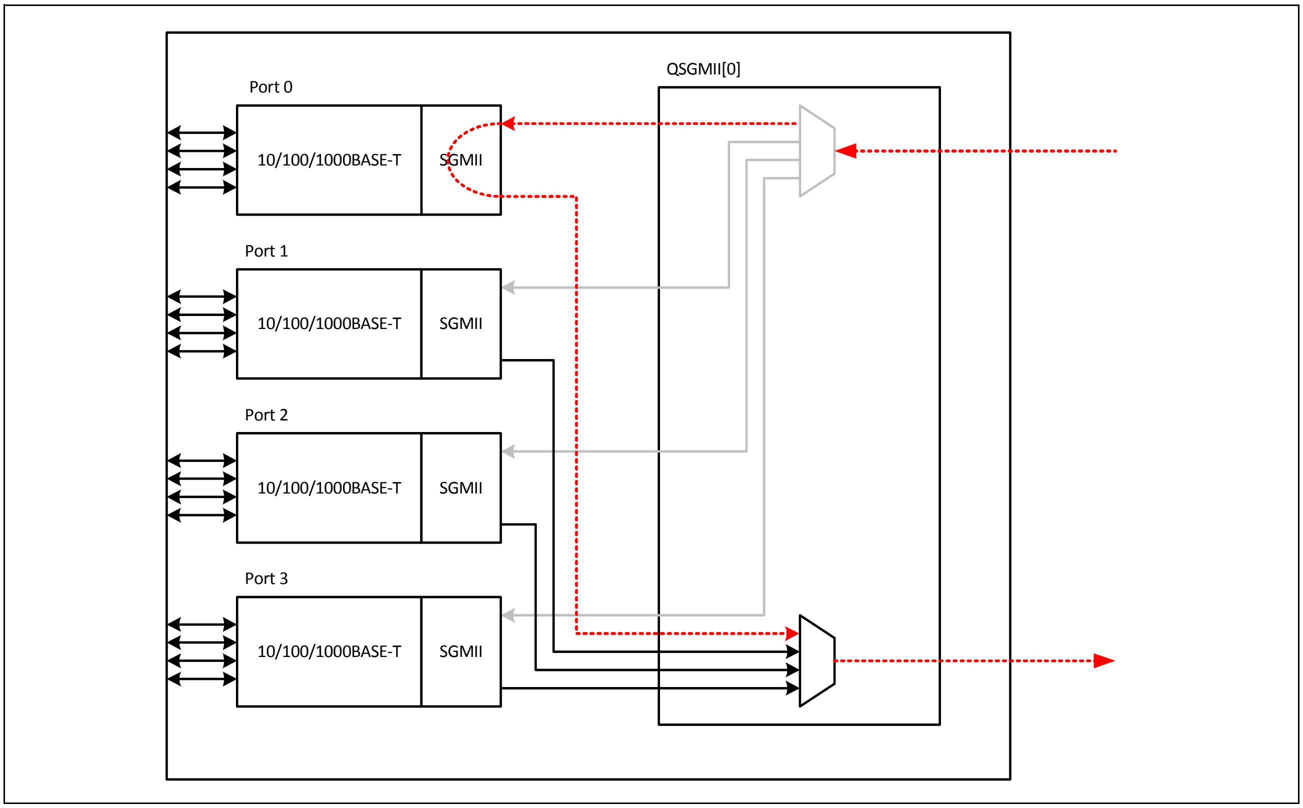


Figure 25: Per Port QSGMII/SGMII Loopback on Port 0

To enable per port QSGMII/SGMII Loopback, do the following writes:

- Write RDB_Register, offset 0x021, bit[0] = 1'b'1 (1000BASE-T register space selected)
- Write RDB_Register, offset 0x00E, bit[12] = 1'b1 (Force link when in 10 Mbps or 100 Mbps mode. Not needed for 1000 Mbps mode.)
- Select the QSGMII speed for the port:
  - Write register 0x0 = 0x4140 to enable 1000 Mbps loopback or
  - Write register 0x0 = 0x6100 to enable 100 Mbps loopback or
  - Write register 0x0 = 0x4100 to enable 10 Mbps loopback

QSGMII PRBS Packet Loopback

This allows 5 Gbs PRBS data to be sent to the PHY’s QSGMII receive input to the PHY’s QSGMII transmit output. This mode is enabled on a per-QSGMII basis. The red dashed line in Figure 26 shows the loopback path for the 5 Gbps PRBS data on QSGMII[0].

Caution! The reference clocks for the PHY and switch must be from the same reference clock source since there is no clock compensation PHY’s FIFO in the path.

Note: To exit per port QSGMII packet loopback, Broadcom recommends a software or hardware reset.

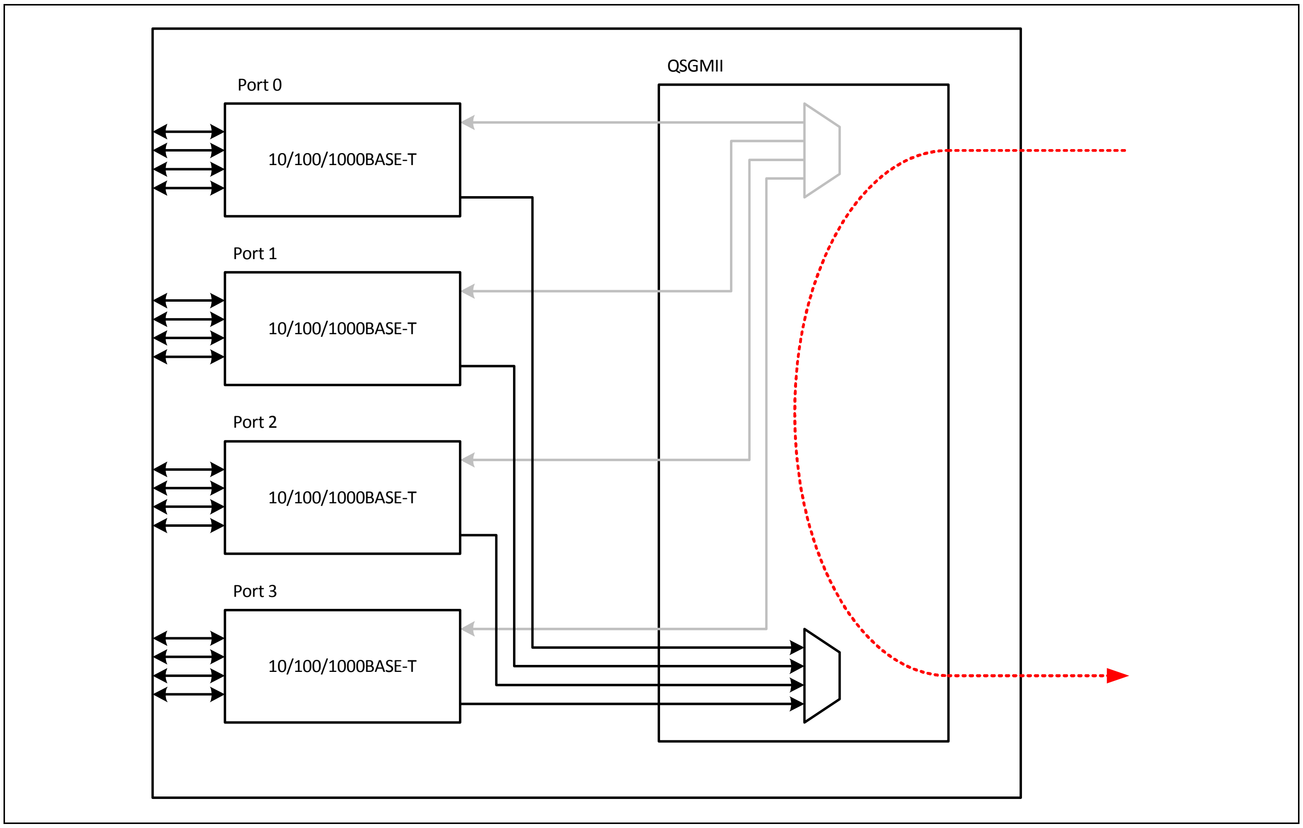


Figure 26: QSGMII 5 Gbs PRBS Loopback

To enable 5 Gbs PRBS loopback, do the following writes:

- Write register 0x1F = 0xFFD0 (set BAR to AER register)
  - Write register 0x1E = 0x0 (enable read/writes to QSGMII[0]) and or
  - Write register 0x1E = 0x4 (enable read/writes to QSGMII[1]) or
  - Write register 0x1E = 0x1F (enable read/writes to QSGMII[0] or QSGMII[1])
- Write QSGMII register (BAR = 0x8000, REGAD = 0x10) = 0x202F (enable 5 Gbs mode)
- Write QSGMII register (BAR = 0x8000, REGAD = 0x1E) = 0x0001 (enable 5 Gbs register space)
- QSGMII register (BAR = 0x0000, REGAD = 0x0) = 0xA040 (soft reset)
- QSGMII register (BAR = 0x8000, REGAD = 0x10) = 0x062F (clear PLL start sequencer)
- QSGMII register (BAR = 0x8300, REGAD = 0x18) = (set PLL to match PHY/switch reference clock)
  - 0x5800 (for 25 MHz or 125 MHz reference clock) or
  - 0x6600 (for 156.25 MHz or 325.5 MHz reference clock)
- QSGMII register (BAR = 0x8000, REGAD = 0x10) = 0x263F (clear PLL start sequencer)
- QSGMII register (BAR = 0x8010, REGAD = 0x15) = 0x0000 (disable QSGMII Clause 36)
- QSGMII register (BAR = 0x8010, REGAD = 0x16) = 0x0303 (set PLL divider to 1)
- QSGMII register (BAR = 0x8010, REGAD = 0x17) = 0x0010 (disable [8b/10b and comma detect], enable loopback)
- QSGMII register (BAR = 0x8010, REGAD = 0x19) = 0x0008 (enable PRBS mode)

Section 9: Timing and AC Characteristics

Reset Timing

Table 37: Reset Timing

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
Power-up to $\overline{\text{RESET}}$ deassertion	RESET_PU	10	–	–	ms
$\overline{\text{RESET}}$ deassertion to normal PHY operation	RESET_WAIT	20	–	–	µs
$\overline{\text{RESET}}$ pulse length	RESET_LEN	2	–	–	µs
$\overline{\text{RESET}}$ rise/fall time	–	–	–	25	ns

Note:

- $\overline{\text{RESET}}$ must be low when power supplies are ramping up.
- When $\overline{\text{RESET}}$ is low, there must be a valid clock signal at the REFCLK input. All external power supplies need to be stable.
- MII register read/write access and normal PHY operation can start at the end of the RESET_WAIT time.
- RESET_PU must be performed when the device is first powered up. Software reset or RESET_LEN does not need to be performed after RESET_PU.
- Software reset or RESET_LEN should not be performed until after RESET_PU and RESET_WAIT have been completed. After issuing a software reset or a RESET_LEN, normal PHY operation can begin after waiting the RESET_WAIT time of 20 µs.

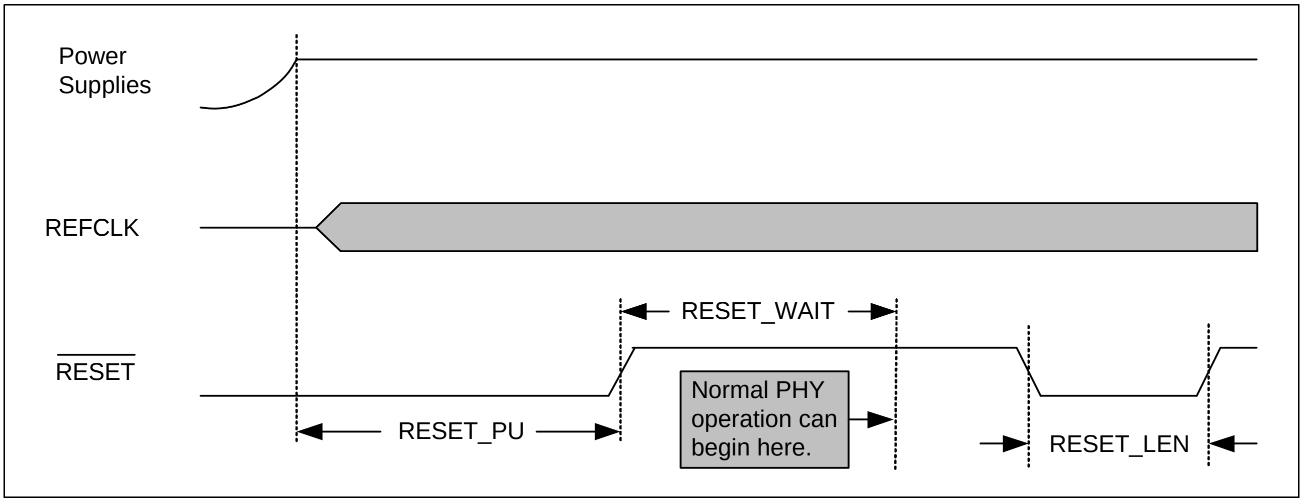


Figure 27: Reset Timing

REFCLK Input Timing (Single-Ended Mode)

Table 38: REFCLK Input Timing

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
Frequency	C_{freq}	–	25	–	MHz
Frequency	Cfreq	–	125	–	MHz
Frequency	Cfreq	–	156.25	–	MHz
Accuracy	–	–50	–	+50	ppm
Duty cycle distortion[a]	–	40	–	60	%
Rise/fall time[b]	T_r/T_f	–	–	1	ns
RMS phase jitter[c]	–	–	–	1.5	ps-rms
Do not use PLL-based oscillators or zero-delay buffers as a source for REFCLK because this introduces excessive jitter that may result in unacceptable bit error rate performance.					

a. Measured at 50% point.
b. Measured at the 20% to 80% points.
c. Frequency = 25 MHz: Fj = 1 kHz to 5 MHz offset frequency.
Frequency > 25 MHz: Fj = 12 kHz to 20 MHz offset frequency.

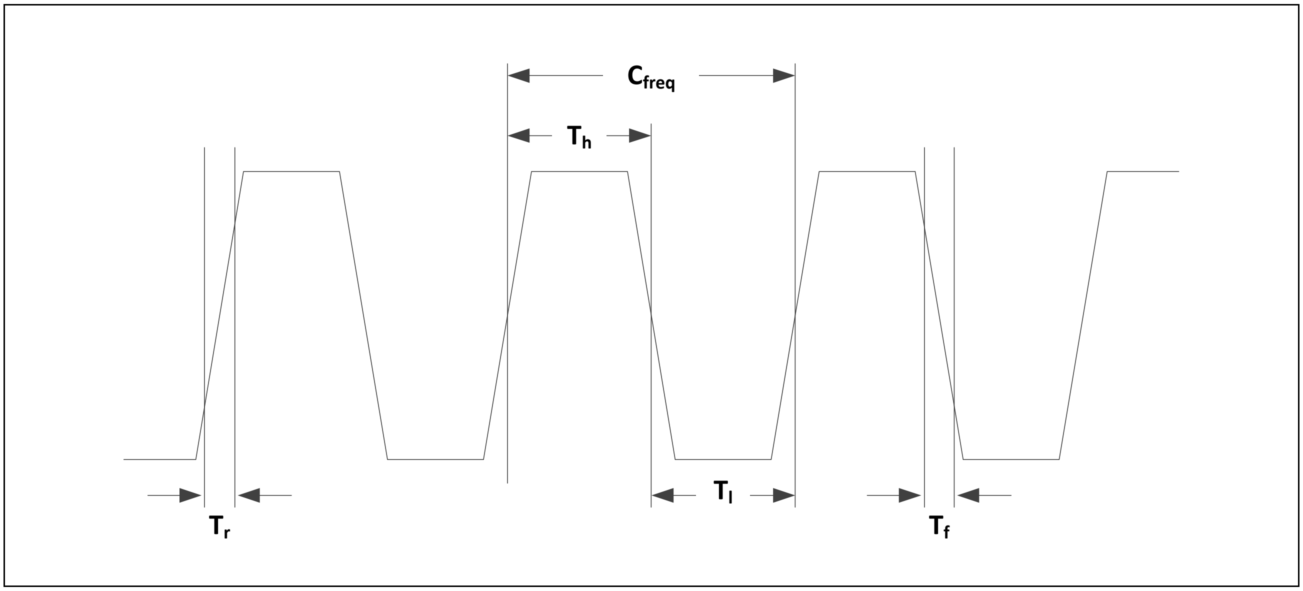


Figure 28: Single-Ended REFCLK Input Timing

REFCLK Clock Input Timing (Differential Mode)

Table 39: REFCLKP/N Clock Input Timing

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
Frequency	C_{freq}	–	25	–	MHz
Frequency	Cfreq	–	125	–	MHz
Frequency	C_{freq}	–	156.25	–	MHz
Frequency	C_{freq}	–	312.5	–	MHz
Frequency deviation	PPM	–50	–	+50	PPM
Duty cycle [a]	T_h/T_l	40	50	60	%
RMS phase jitter[b]	–	–	–	1.5	ps-rms
Rise/fall time[c]	T_r/T_f	–	–	1	ns
Differential skew[d]	T_{skew}	–	–	80	ps

a. Measured at 50% crossing-point.
b. Frequency = 25 MHz: Fj = 1 kHz to 5 MHz offset frequency.
Frequency > 25 MHz: Fj = 12 kHz to 20 MHz offset frequency.
c. Measured at 20% - 80% points.
d. Skew between two members of a differential pair measured at 50% crossing-point.

Note: REFCLKP/N must be AC-coupled to the clock source.

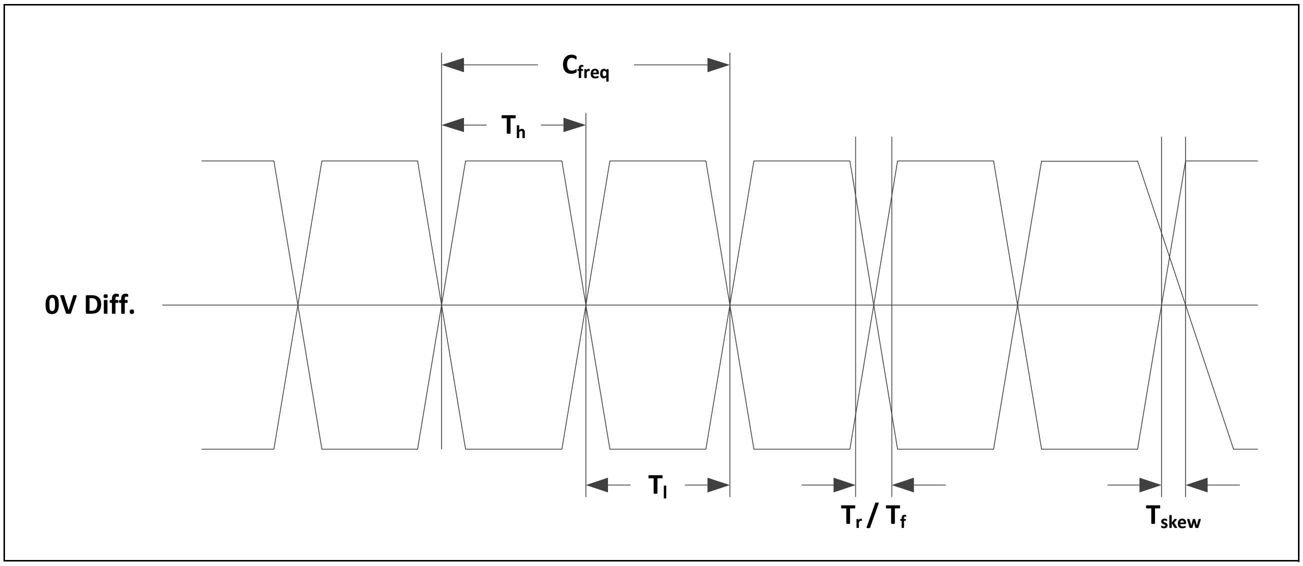


Figure 29: Differential REFCLK Input Timing

Management Interface Timing

Table 40: Management Interface Timing

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
MDC cycle time	MDC_CYCLE	80	–	–	ns
		–	–	12.5	MHz
MDC duty cycle	MDC_HI/LOW	30	–	70	%
MDIO input setup time to MDC rising	MDIO_SETUP	5	–	–	ns
MDIO input hold time from MDC rising	MDIO_HOLD	5	–	–	ns
MDIO output delay from MDC rising	MDIO_DELAY				
OVDDMDIO = 3.3V or 2.5V		5	–	15	ns
OVDDMDIO = 1.2V		5	–	50	ns

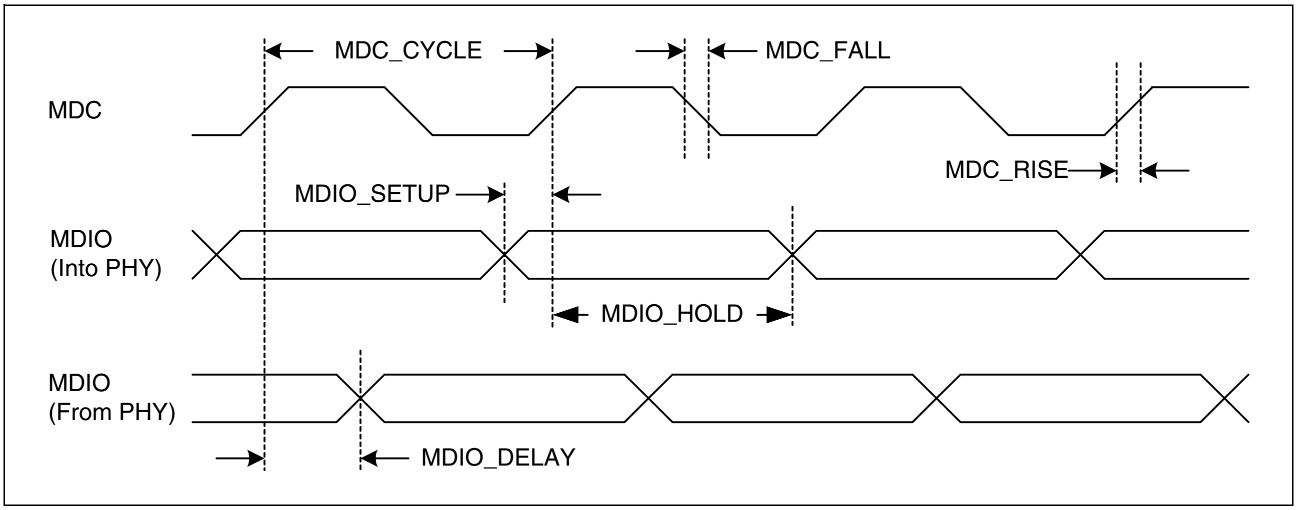


Figure 30: Management Interface Timing

JTAG Timing

Table 41: JTAG Timing

Parameter	*Symbol*	*Minimum*	*Maximum*	*UnIt*
TCK Period	T_period	50	–	ns
TCK High Time	T_hi	15	–	ns
TCK Low Time	T_low	15	–	ns
TDI/TMS Set-up Time	T_setup	10	–	ns
TDI/TMS Hold Time	T_hold	5	–	ns
TCK to TDO Delay (5pf load)	T_delay	4.5	20	ns

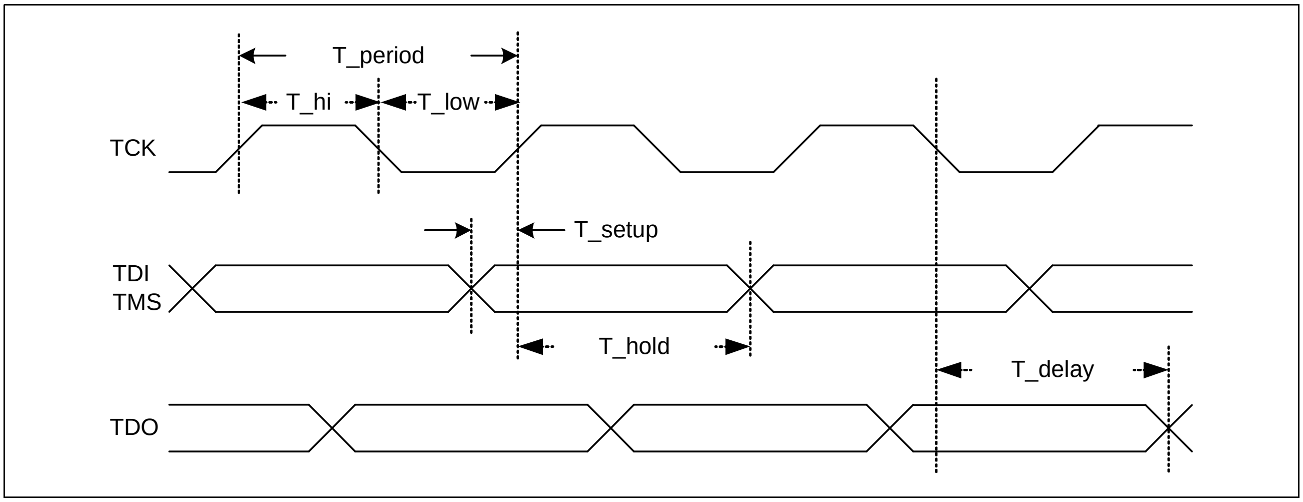


Figure 31: JTAG Timing

Section 10: Electrical Characteristics

Absolute Maximum Ratings

Table 42: Absolute Maximum Ratings

Parameter	***Minimum***	***Maximum***	***Unit***
1.0V Supplies AVDDL, DVDD, PLLVDD, SPLLVDD, QRVDD, QTVDD, QPVDD, SAVDDR, SAVDDT	GND – 0.5	1.37	V
1.2V Supplies OVDDMDI0	GND – 0.5	1.37	V
1.8V Supplies OVDDJTAG	GND – 0.5	2.10	V
2.5V Supplies OVDDMDIO, OVDDJTAG	GND – 0.5	3.10	V
3.3V Supplies AVDDH, BIASVDD, OVDD, OVDDMDIO, OVDDJTAG, CLKVDD	GND – 0.5	4.10	V
Digital input overshoot: < 700 mV above power rail for less than 3 ns. Overshoot: For any 3.3V I/O, the absolute maximum overshoot needs to be less than 4.1V. Recommend limiting the I/O overshoot to be less than 500 mV above OVDD supply for no more than 5% duty cycle.	OVDD + 0.700	–	V
Digital input undershoot: < 700 mV below ground for less than 3 ns. Undershoot: If the I/O undershoot is less than 500 mV below ground, there is no duration timing requirement. If the I/O undershoot is less than 700 mV below ground, it needs to be less than 3 ns duration.	GND – 0.700	–	V
Storage temperature	–40	125	°C
Note: These specifications indicate levels where permanent damage to the device can occur. Functional operation is not guaranteed under these conditions. Operation at absolute maximum conditions for extended periods can adversely affect long-term reliability of the device.			

DC Characteristics

Power Parameters

Table 43: Recommended Power Supply Voltage Levels

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
Supply voltage 1.0V	V_{AVDDL} V_{DVDD} V_{PLLVDD} $V_{SPLLVDD}$ V_{SAVDDR} V_{SAVDDT} V_{QTVDD} V_{QRVDD} V_{QPVDD}	0.95	1.0	1.05	V
Supply voltage 1.2V (3.3V tolerant)	$V_{OVDDMDIO}$	1.14	1.2	1.26	V
Supply voltage 1.8V (3.3V tolerant)	$V_{OVDDJTAG}$	1.71	1.8	1.89	V
Supply voltage 2.5V (3.3V tolerant)	$V_{OVDDMDIO}$ $V_{OVDDJTAG}$	2.375	2.5	2.625	V
Supply voltage 3.3V	V_{OVDD} V_{AVDDH} $V_{BIASVDD}$ V_{CLKVDD} $V_{OVDDMDIO}$ $V_{OVDDJTAG}$	3.135	3.3	3.465	V

Table 44: QSGMII to Copper Current Consumption

Parameter	Symbol	Minimum	Typical	Maximum	Unit
3.3V digital supply current	I_{OVDD} $I_{OVDDMDIO}$ $I_{OVDDJTAG}$	–	2	–	mA
3.3V analog supply current	I_{AVDDH} $I_{BIASVDD}$ I_{CLKVDD}	–	427	–	mA
1.0V digital supply current	I_{DVDD}	–	485	–	mA
1.0V analog supply current	I_{AVDDL} I_{PLLVDD} I_{QTVDD} I_{QRVDD} I_{QPVDD} $I_{SPLLVDD}$ I_{SAVDDL} I_{SAVDDR}	–	395	–	mA
Total power	P_{Total}	–	2296	–	mW
Power per port	P_{PP}	–	287	–	mW

Note: Over process, voltage and temperature, the maximum power dissipation is approximately 15% to 20% higher than the typical value.

Table 45: QSGMII to Fiber Current Consumption

Parameter	Symbol	Minimum	Typical	Maximum	Unit
3.3V digital supply current	I_{OVDD} $I_{OVDDMDIO}$ $I_{OVDDJTAG}$	–	2	–	mA
3.3V analog supply current	I_{AVDDH} $I_{BIASVDD}$ I_{CLKVDD}	–	62	–	mA
1.0V digital supply current	I_{DVDD}	–	289	–	mA
1.0V analog supply current	I_{AVDDL} I_{PLLVDD} I_{QTVDD} I_{QRVDD} I_{QPVDD} $I_{SPLLVDD}$ I_{SAVDDL} I_{SAVDDR}	–	401	–	mA
Total power	P_{Total}	–	902	–	mW
Power per port	P_{PP}	–	113	–	mW
Note: Over process, voltage and temperature, the maximum power dissipation is approximately 15% to 20% higher than the typical value.					

I/O Parameters

Table 46: I/Os Operating at OVDD = 3.3V

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input high-voltage	V_{IH}	2.00	–	V	–
Input low-voltage	V_{IL}	–	0.80	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	$I_{OH} = -8$ mA
Output low-voltage	V_{OL}	–	0.40	V	$I_{OL} = 8$ mA

MDIO/MDC Parameters

Table 47: MDIO/MDC: OVDDMDIO Operating at 3.3V

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input high -voltage	V_{IH}	2.00	–	V	–
Input low-voltage	V_{IL}	–	0.80	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

Table 48: MDIO/MDC: OVDDMDIO Operating at 2.5V

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input high-voltage	V_{IH}	1.70	–	V	–
Input low-voltage	V_{IL}	–	0.70	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

Table 49: MDIO/MDC: OVDDMDIO Operating at 1.2V

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input high-voltage	V_{IH}	0.84	–	V	–
Input low-voltage	V_{IL}	–	0.36	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

JTAG Parameters

Table 50: JTAG OVDDJTAG Operating at 3.3V

Parameter	Symbol	Minimum	Maximum	Unit	Condition
Input high-voltage	V_{IH}	2.00	–	V	–
Input low-voltage	V_{IL}	–	0.80	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

Table 51: JTAG: OVDDJTAG Operating at 2.5V

Parameter	*Symbol*	*Minimum*	*Maximum*	*Unit*	*Condition*
Input high-voltage	V_{IH}	2.00	–	V	–
Input low-voltage	V_{IL}	–	0.70	V	–
Output high-voltage	V_{OH}	OVDD – 0.45	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

Table 52: JTAG: OVDDJTAG Operating at 1.8V

Parameter	*Symbol*	*Minimum*	*Maximum*	*Unit*	*Condition*
Input high-voltage	V_{IH}	1.22	–	V	–
Input low-voltage	V_{IL}	–	0.60	V	–
Output high-voltage	V_{OH}	OVDD – 0.40	–	V	I_{OH} = –8 mA
Output low-voltage	V_{OL}	–	0.40	V	I_{OL} = 8 mA

Reference Clock Parameters

Table 53: Single-Ended Reference Clock

Parameter	*Symbol*	*Minimum*	*Maximum*	*Unit*	*Condition*
Input high-voltage	V_{IH}	2.1	–	V	–
Input low-voltage	V_{IL}	–	0.70	V	–

Table 54: CML Differential Reference Clock

Parameter	*Symbol*	*Minimum*	*Maximum*	*Unit*	*Condition*
Reference clock input voltage swing differential	$V_{pk\text{-}pk}$	600	2000	mVppd	–
Differential input impedance	R_{IN}	90	130	ohms	–
Duty cycle	%	40	60	%	–

QSGMII Parameters

Table 55: QSGMII Transmitter

Parameter	Symbol	Minimum	Typical	Maximum	Unit	Conditions
Baud rate	T_{Baud}	–	5.000	–	Gsym/s	–
Output differential voltage[a]	T_{DIFF}	400	–	1200	mVppd	Floating load R_{load} = 100Ω
Differential resistance	T_{RD}	80	100	120	Ω	–
Output rise and fall times	T_R, T_F	30	–	–	ps	20% to 80%
Differential output return loss	T_{SDD22}	–	–	–8	dB	100 MHz to 2.5 GHz
		–	–	–	dB	2.5 GHz to 5 GHz
Common mode return loss	T_{SCC22}	–	–	–6	dB	100 MHz to 2.5 GHz
Output common mode noise	TN_{CM}	–	–	5% of T_{DIFF}	mVppd	–
Output current short	T_{IS}	–	–	100	mA	Output shorted to gnd or to each other.
Output common mode voltage	T_{CM}	550		1060	mV	Load Type 2[b]

a. Output differential voltage is programmable through bits[10:6] in QSGMII Address:
BAR = 0x8060, REGAD = 0x17, AER = 0x0000 for QSGMII[0] and AER = 0x0004 for QSGMII[1]

b. Load Type 2 specified per the IA # OIF-CEI-02.0 specification.

Table 56: QSGMII Transmit Jitter

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Uncorrelated high-probability jitter	T_{UHPJ}	–	–	0.15	UIpp
Duty cycle distortion	T_{DCD}	–	–	0.05	UIpp
Total jitter	T_{TJ}	–	–	0.30	UIpp
Eye mask	T_{X1}	–	–	0.15	UI
	T_{X2}	–	–	0.40	UI
	T_{Y1}	200	–	–	mV
	T_{Y2}	–	–	450	mV

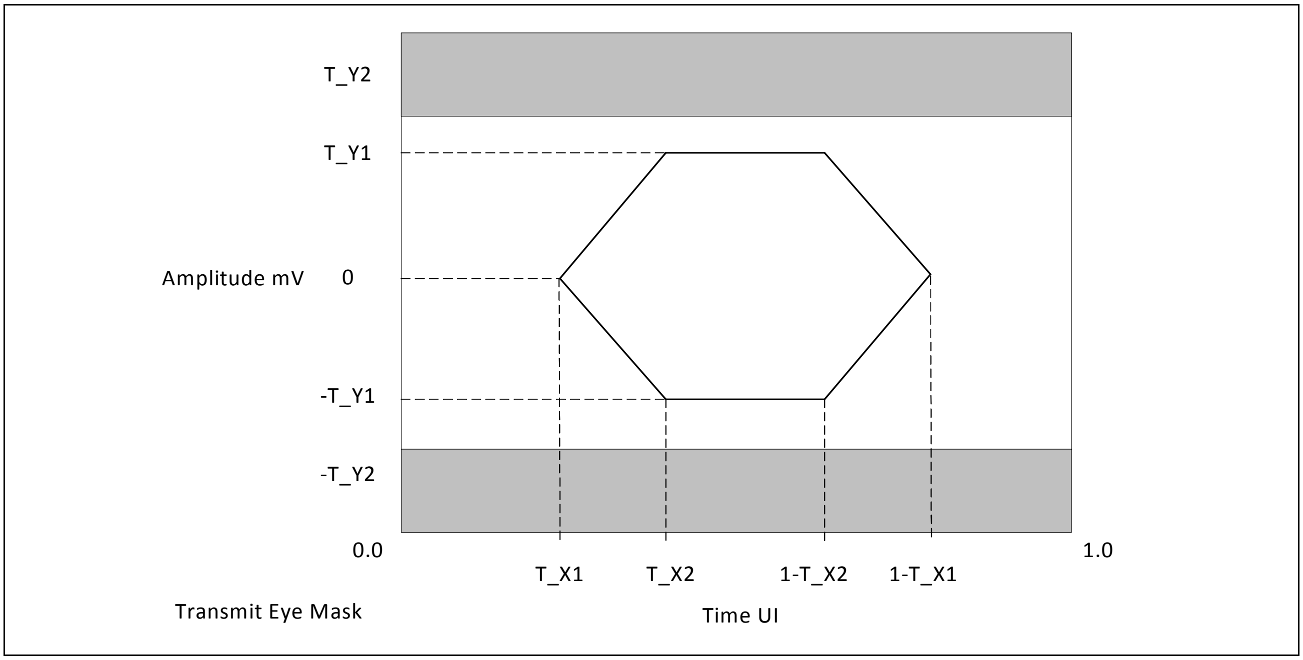


Figure 32: QSGMII Transmit Eye Mask

Table 57: QSGMII Receiver

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*	*Notes*
RX baud rate	R_{BAUD}	–	5.0	–	Gsym/s	–
Input differential voltage	R_{DIFF}	100	–	900	mVppd	–
Differentia resistance	R_{RDIN}	80	100	120	Ω	Load Type 2[a]
Bias voltage source impedance	R_{ZVTT}	–	–	30	Ω	Load Type 2
Differential input return loss	R_{SDD1}	–	–	30	Ω	100 MHz to 2.5 GHz
		–	–	–8	dB	2.5 GHz to 5 GHz
Common mode input return loss	R_{SCC1}	–	–	–6	dB	100 MHz to 2.5 GHz
		Loss dB = A0+16.6*log(f/2.5 GHz)			dB	f = 2.5 GHz to 5.0 GHz
Termination voltage	R_{VTT}	1.0 – 8%	–	1.0 + 5%	V	R_{VTT} = 1.0V nominal Load Type 2
Input common mode voltage	R_{CM}	535	–	R_{VTT} + 125 mV	mV	R_{VTT} = 1.0V nominal Load Type 2
Wander divider	n	–	–	10	–	–

[a] Load Type 2 specified per the IA # OIF-CEI-02.0 specification.

Table 58: QSGMII Receive Jitter

Parameter	*Symbol*	*Minimum*	*Typical*	*Maximum*	*Unit*
Bounded high-probability jitter	R_{BHPJ}	–	–	0.45	UIpp
Sinusoidal jitter, maximum	$R_{SJ\text{-}max}$	–	–	5	UIpp
Sinusoidal jitter, high frequency	$R_{SJ\text{-}hf}$	–	–	0.05	UIpp
Total jitter (does not include sinusoidal jitter)	R_{TJ}	–	–	0.60	UIpp
Eye mask	R_{X1}	–	–	0.30	UI
	R_{Y1}	200	–	50	mV
	R_{Y2}	–	–	450	mV

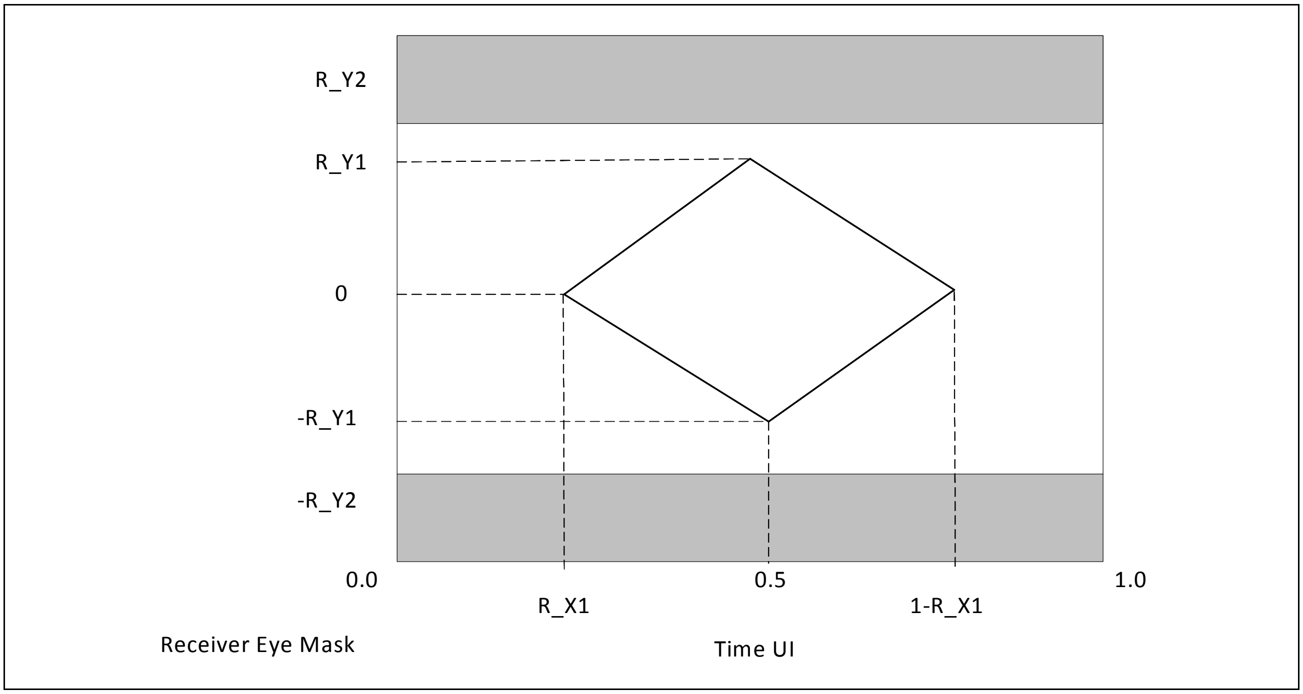


Figure 33: QSGMII Receive Eye Mask

Section 11: Thermal Information

Package

This section includes thermal information for the BCM54285 package. Thermal numbers are based on JEDEC standards for 2s2p PCB (4 layer PCB board), wind tunnel and still air enclosures, therefore the temperatures might be different in the customers environment. Table 59 provides Theta-JB and Theta-JC data. Table 60 provides Theta-JA and PSI-JT thermal data.

Note: Maximum steady state die junction temperature in a stable environment of device ambient, airflow or heat sink is 110°C.

Note: Maximum excursion die junction temperature due to the loss of airflow or heat sink integrity or due to significant fluctuation of device ambient temperature or other events is 125°C. The duration of the excursion should be less than 1000 hours for the lifetime of the product.

Table 59: Theta-JB and Theta-JC

Parameter	***Value***	***Units***
θ_{JB}	11.10	°C/W
θ_{JC}	9.07	°C/W

Table 60: Theta-JA and PSI-JT

Air Velocity (m/s)	***0***	***0.508***	***1.016***	***2.032***	***3.048***	***Condition***
Air Velocity (ft/Min)	***0***	***100***	***200***	***400***	***600***	
θ_{JA} (°C/W)	25.48	23.76	22.83	21.90	21.31	No heat sink
Ψ_{JT} (°C/W)	2.62	3.14	3.41	3.79	4.07	
θ_{JA} (°C/W)	13.10	11.03	10.59	10.24	10.17	External heat sink[a]
Ψ_{JT} (°C/W)	8.48	8.55	8.59	8.59	8.62	

[a] Heat sink: 60 mm x 60 mm x 30 mm fin type heat sink, aluminum

Thermal Data Definitions

The thermal models for package Junction-to-Board (Theta-JB) and Junction-to-Case (Theta-JC) thermal resistances are constructed based on generic thermal resistance definitions. The simulated θ_{JB} and θ_{JC} values depend on package internal construction only. They are not dependent on PCB used in either tests or applications. The simulated θ_{JB} and θ_{JC} values provide package two-resistor compact model. The following are the equations for θ_{JA}, θ_{JB}, and θ_{JC}:

$$\theta_{JA} = \frac{T_J - T_A}{P}$$

$$\theta_{JB} = \frac{T_J - T_B}{P}$$

$$\theta_{JC} = \frac{T_J - T_C}{P}$$

Where:

- T_J = Junction temperature at steady-state condition. Units = °C.
- T_A = Package case top center temperature at steady-state condition. Units = °C.
- T_B = Package lead footprint temperature specified in thermal simulation. Units = °C.
- T_C = package case top surface temperature specified in thermal simulation. Units = °C.
- P = Device power dissipation. Units = Watts.

Junction Temperature Estimation and PSI$_{JT}$ Versus Theta$_{JC}$

Package thermal characterization parameter Psi-J$_T$ (Ψ_{JT}) yields a better estimation of actual device junction temperature (T_J) versus using the junction-to-case thermal resistance parameter Theta-J$_C$ (θ_{JC}). The reason for this is θ_{JC} assumes that all the power is dissipated through the top surface of the package case. In actual applications, some of the power is dissipated through the bottom and sides of the package. Ψ_{JT} takes into account power dissipated through the top, bottom and sides of the package. The equation for calculating the device junction temperature is as follows.

$$T_J = T_T + P \times \Psi_{JT}$$

Where:

- T_J = Junction temperature at steady-state condition. Units = °C.
- T_T = Package case top center temperature at steady-state condition. Units = °C.
- P = Device power dissipation. Units = Watts.
- Ψ_{JT} = Package thermal characteristics. Units = °C/W.

RoHS-Compliant Packaging

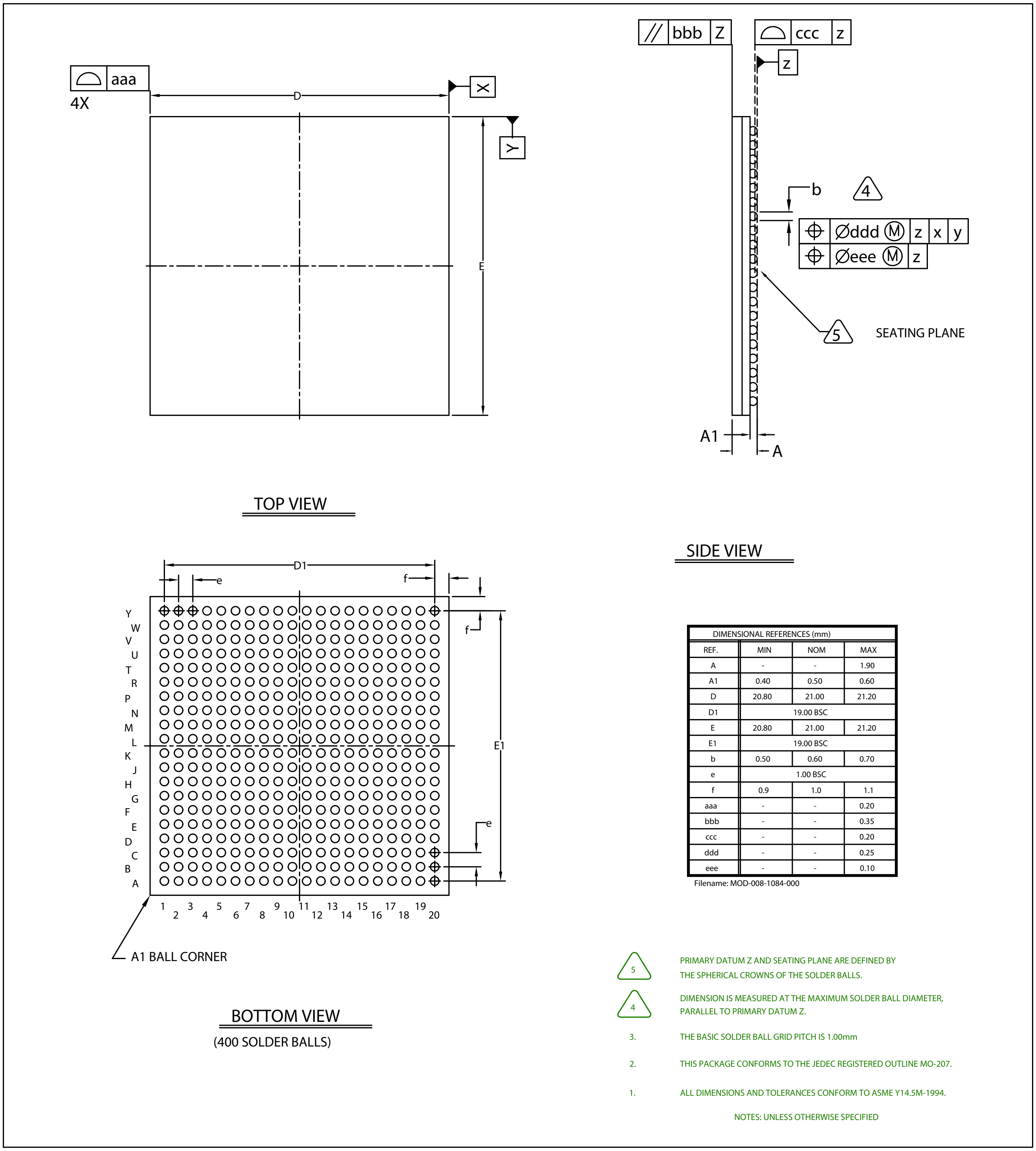
Broadcom’s RoHS packages are in compliance with RoHS and WEEE directives. Broadcom may also offer standard parts that are also in compliance with these directives, with the exception of Pb (>1000 ppm). Table 61 shows the main differences between standard and RoHS-compliant parts.

RoHS-compliant parts have a letter 'G' added to the top line of the part marking. Standard parts (non Pb-free parts) are not compatible to Pb-free surface mount process. See Section 13: “Ordering Information,” on page 125 and see the application note *Reflow Process Guidelines for Surface Mount Assemblies* (PACKAGING-AN10x-R).

Table 61: Main Differences Between Standard and RoHS-Compliant Packages

Package	*Solder Ball Composition*	*Maximum Reflow Temperature (°C)*
Standard package	63%Sn/37%Pb	225
RoHS-compliant package	96.5%Sn/3%Ag/0.5%Cu	255

Section 12: Mechanical Information



DIMENSIONAL REFERENCES (mm)			
REF.	MIN	NOM	MAX
A	-	-	1.90
A1	0.40	0.50	0.60
D	20.80	21.00	21.20
D1	19.00 BSC		
E	20.80	21.00	21.20
E1	19.00 BSC		
b	0.50	0.60	0.70
e	1.00 BSC		
f	0.9	1.0	1.1
aaa	-	-	0.20
bbb	-	-	0.35
ccc	-	-	0.20
ddd	-	-	0.25
eee	-	-	0.10

Filename: MOD-008-1084-000

5 PRIMARY DATUM Z AND SEATING PLANE ARE DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.

4 DIMENSION IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER, PARALLEL TO PRIMARY DATUM Z.

3. THE BASIC SOLDER BALL GRID PITCH IS 1.00mm

2. THIS PACKAGE CONFORMS TO THE JEDEC REGISTERED OUTLINE MO-207.

1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ASME Y14.5M-1994.

NOTES: UNLESS OTHERWISE SPECIFIED

Figure 34: 400-Ball FBGA Package

Section 13: Ordering Information

Table 62: Ordering Information

Part Number	*Package*	*Ambient Temperature*
BCM54285C1KFBG	400-ball FBGA (RoHS-compliant)	0°C to 70°C

Appendix A: Acronyms and Abbreviations

The following lists of acronyms and abbreviations applies specifically to Broadcom and associated products. These lists are updated regularly. If a term is not listed, but should be, contact a Broadcom representative.

For a complete list of acronyms and other terms used in Broadcom documents, go to: http://www.broadcom.com/press/glossary.php.

Term	*Description*
ADC	Analog-to-digital converter
APD	Auto Power-Down
BER	Bit error rate
BSC	Broadcom Serial Control
CF	Correction fields
CLI	Command Line Interface
CSP	Customer Support Portal
DAC	Digital-to-analog converter
DPLL	Digital Phase-Locked Loop
DFE	Decision-Feedback Equalization
EEE	Energy Efficient Ethernet
EOL	End-of-life
EMI	Electromagnetic interference
FEC	Forward error correction
FFE	Feed Forward Equalization
ISI	Intersymbol interference
LFSR	Linear Feedback Shift Register
LLDP	Link Layer Discovery Protocol
GM	Grand Master
GUI	Graphical user interface
LPI	Low-power idle
MDI	Media Dependent Interface
NSE	Network Synchronization Engine
NEXT	Near-end crosstalk
NRZ	Nonreturn to zero
PDU	Protocol data unit
SC	Slave Clock
RX_ER	Receive Error
SFD	Start Frame Delimiter

Term	***Description***
SoC	System-on-a-Chip
TC	Transparent Clock
WDT	Watchdog timer

Connecting everything®

BROADCOM CORPORATION
5300 California Avenue
Irvine, CA 92617

Phone: 949-926-5000
Fax: 949-926-5203
E-mail: info@broadcom.com
Web: www.broadcom.com