Core10GBaseKR_PHY User Guide

Introduction (Ask a Question)

The Microchip FPGA Core10GBaseKR_PHY core is designed in accordance with the IEEE® 802.3 2012 standard and supports Core10GBaseKR_PHY interface for the Backplane operations. This configurable core provides the Physical (PHY) layer when used with a transceiver interface. This IP interfaces with the Ten Gigabit Media Independent Interface (XGMII) compliant Media Access Control (MAC) at the system side and the transceiver block at the Line side. The physical layer is designed to work seamlessly with the PolarFire® and PolarFire® SoC transceiver using the Physical Medium Attachment (PMA) mode.

This user guide is a part of the production release of Core10GBaseKR_PHY v3.0. This document provides information on how to interface the transceiver and the XGMII compliant MAC using the Core10GBaseKR_PHY for Backplane applications in PolarFire and PolarFire SoC devices. For additional information, see PolarFire FPGA and PolarFire SoC FPGA Transceiver User Guide and AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note .

Important: The Advanced Peripheral Bus (APB) protocol standard uses the terminology Master and Slave. The equivalent Microchip terminology used in this document is **Initiator** and **Target**, respectively.

Summary

Core Version	This document applies to Core10GBaseKR_PHY v3.0.
Supported Device Families	• PolarFire® • PolarFire SoC
Supported Tool Flow	Requires Libero® v12.6 or later .
Supported Interfaces	APB Slave interface
Licensing	This core supports generation of obfuscated Verilog version of the core. Obfuscated version license locked at packaging time and require the presence of a FLEXlm license to generate: **Evaluation:** Evaluation version is available with encrypted Verilog RTL with self-destruct logic. Evaluation version works approximately four hours of functionality on silicon. Evaluation version is not license locked and available for free. **Obfuscated:** Obfuscated version is available with encrypted Verilog RTL, Obfuscated version supports unlimited functionality on silicon. Obfuscated version is license locked. You must purchase this license separately.

Features

Core10GBaseKR_PHY has the following key features:

- 64-bit XGMII interface towards the MAC side and 32-bit PMA interface
- Programmable IEEE Clause 73 (Auto-Negotiation) and Clause 72 (Link Training)
- APB interface for the register access
- 64B/66B Physical Coding Sublayer (PCS) encoding or decoding as per the IEEE Clause 49

- Elastic Buffer implemented in the Receive path to absorb the +/-100 ppm frequency variation. This is achieved by addition or deletion of the skip characters in the Ethernet traffic.
- FEC capability as per the IEEE 802.3-2012 specification

Installation Instructions

Core10GBaseKR_PHY must be installed to the IP Catalog of Libero® SoC software automatically through the IP Catalog update function in Libero SoC software, or it can be manually downloaded from the catalog. Once the IP core is installed in Libero SoC software IP Catalog, it can be configured, generated, and instantiated within SmartDesign for inclusion in the Libero project.

Device Utilization and Performance (Ask a Question)

Core10GBaseKR_PHY has been implemented in the PolarFire MPF300TS device using speed grade -1 and PolarFire SoC MPFS250TS device using speed grade -1.

Resource utilization and performance data in the following tables are achieved by applying timing, synthesis, and place-and-route constraints. For more information, see 6.1. Timing Constraint , 4.1.2. Synthesis in Libero SoC, and 4.1.3. Place-and-Route in Libero SoC. Modify the default values of the Core10GMAC parameters such as **10G Type**, **System Data Width**, and **Core Data Width**. The following tables list the resource utilization and performance value for the Core10GBaseKR_PHY IP.

Table 1. Core10GBaseKR_PHY Utilization when FEC is Disabled

Device Details			Resources			Utilization	Performance (MHz)	µSRAM
Family	Device	Speed Grade	4-LUT	DFF	Logic Elements	Total%		
PolarFire®	MPF300TS	−1	8411	7887	10290	3.44	I_XGMII_TX_CLK – 240 I_XGMII_RX_CLK – 250 TX_CLK_R – 342 RX_CLK_R – 340 PCLK – 80	14
PolarFire SoC	MPFS250TS	−1	8409	7883	10315	4.06	I_XGMII_TX_CLK – 240 I_XGMII_RX_CLK – 252 TX_CLK_R – 340 RX_CLK_R – 335 PCLK – 80	14

Table 2. Core10GBaseKR_PHY Utilization when FEC is Enabled

Device Details			Resources			Utilization	Performance (MHz)	µSRAM
Family	Device	Speed Grade	4-LUT	DFF	Logic Elements	Total%		
PolarFire®	MPF300TS	−1	11500	10506	13975	4.67	I_XGMII_TX_CLK – 230 I_XGMII_RX_CLK – 220 TX_CLK_R – 334 RX_CLK_R – 328 PCLK – 80	19
PolarFire SoC	MPFS250TS	−1	11354	10400	13763	5.41	I_XGMII_TX_CLK – 230 I_XGMII_RX_CLK – 220 TX_CLK_R – 330 RX_CLK_R – 328 PCLK – 80	19

Important:

- Supported only with −1 speed
- For more information, see PolarFire FPGA and PolarFire SoC FPGA Transceiver User Guide and AN4592: PolarFire FPGA Dynamic Reconfiguration Interface Application Note

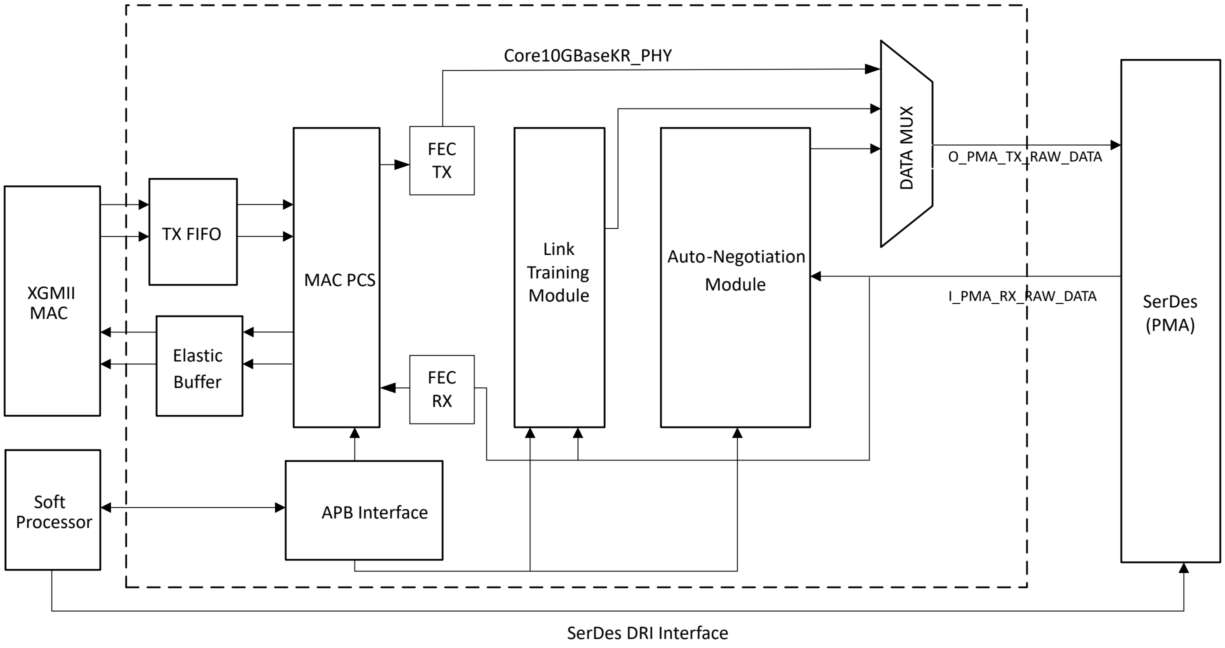
Table of Contents

1. Functional Description (Ask a Question)

Core10GBaseKR_PHY interfaces XGMII compliant MAC with the 10GBaseKR_PHY device for the Backplane applications. The Core10GBaseKR_PHY IP designed in accordance with the *IEEE 802.3-2012 specification*, and supports the XGMII interface towards the MAC side and the PMA interface towards the SerDes side. For 10GBaseKR_PHY applications, the transceiver is used in PMA native mode (32-bit only) and connected to the Ethernet MAC through the PCS interface.

For 10GBaseKR_PHY configuration, Link Training, and Auto-Negotiation module are enabled and can be accessed from the 32-bit APB target interface. The provision to disable the Auto-Negotiation and Link Training module is provided through parameter static configuration as part of this IP. The Core has FEC capability as per the *IEEE 802.3-2012 specification* and the provision to Disable FEC is provided through parameter static configuration. The following figure shows the top-level block diagram of Core10GBaseKR_PHY integrated for the 10GBaseKR_PHY solution.

Figure 1-1. Core10GBaseKR_PHY System-Level Diagram



1.1 Core10GBaseKR_PHY Functional Blocks (Ask a Question)

Core10GBaseKR_PHY contains the following functional blocks:

- PCS Module
- Auto Negotiation
- Link Training
- APB Target Interface for Register Configuration
- XGMII Interface Towards the MAC
- FEC TX and RX

1.1.1 PCS Module (Ask a Question)

The MAC PCS module supports the following features:

- PCS sublayer for 64B/66B encoding and decoding in compliance to the *IEEE 802.3 Clause 49*
- 32-bit data path connection to the transceiver interface

Important: Link Partner and remote receiver words are used interchangebly.

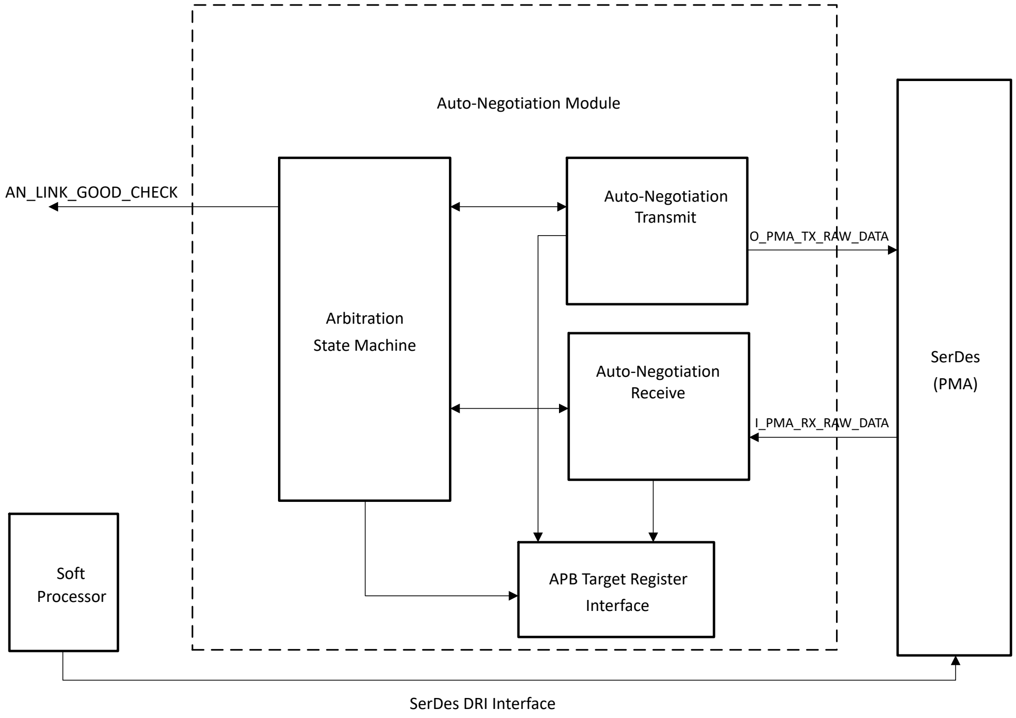
1.1.2 Auto-Negotiation (AN) Module (Ask a Question)

The Auto-Negotiation module supports the ability to determine if the link is 10GBase-KR. Auto-Negotaition Intellectual Property Core (ANIPC) implements the requirement as specified in *Clause 73, IEEE 802.3* standard and supports the following features:

- Transmit State Machine
- Receive State Machine
- Arbitration State Machine

The following figure shows the top-level block diagram of the Auto-Negotiation module.

Figure 1-2. Top-Level Block Diagram of the Auto-Negotiation Module



For a local device, the primary function of the Auto-Negotiation module is to communicate with a link partner through the exchange of DME pages. Each DME page is 48 bits in length. For more information on DME encoding, see *IEEE 802.3 73.5.2 specification*. The following figure shows the link codeword transmitted in the base page.

Figure 1-3. Link Codeword

D 0	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9	D 10	D 11	D 12	D 13	D 14	D 15
S 0	S 1	S 2	S 3	S 4	E 0	E 1	E 2	E 3	E 4	C 0	C 1	C 2	RF	Ack	NP

D 16	D 17	D 18	D 19	D 20	D 21	D 22	D 23	D 24	D 25	D 26	D 27	D 28	D 45	D 46	D 47
T 0	T 1	T 2	T 3	T 4	A 0	A 1	A 2	A 3	A 4	A 5	A 6	A 7	A 24	F 0	F 1

D [4:0]	Selector field
D [9:5]	Echoed Nonce field
D [12:10]	Capability bits to advertise capabilities not related to the PHY
C [1:0]	Advertise pause capability
C [2]	Capablity bit is reserved
D [15:13]	RF, Ack, and NP bits. See *73.6.7, 73.6.8, and 73.6.9 sections of IEEE 802.3 specification*
D [20:16]	Transmitted Nonce field
D [45:21]	Technology Ability field
D [47:46]	FEC capability (see *IEEE 73.6.5 specification*)

1.1.2.1 Auto-Negotiation Transmit (Ask a Question)

The Auto-Negotiation Transmit module provides the ability to transmit pages. The first pages exchanged by the local device and its link partner after power-on, link restart, or renegotiation contain the base link codeword. The Transmit State Machine is implemented in accordance with the *Figure 73-9* as specified in *Clause 73 of IEEE 802.3 standard*.

1.1.2.2 Auto-Negotiation Receive (Ask a Question)

The Auto-Negotiation Receive module detects the DME page sequence, decodes the information, and stores the data in `rx_link_code_word [48:1]`. The Auto-Negotiation Receive module interfaces to the SerDes (PMA) through a 32-bit interface. The Auto-Negotiation RX State Machine is implemented in accordance with the *Figure 73-10* as specified in *Clause 73 of IEEE 802.3 standard*.

Important: The Auto-Negotiation is performed at a 312.5 Mbps line rate. The clocking used is 9.765 MHz for the Auto-Negotiation operation at the receiver side.

1.1.2.3 Arbitration State Machine (Ask a Question)

The Arbitration State Machine function ensures proper sequencing of the Auto-Negotiation function using the Auto-Negotiation Transmit and the Auto-Negotiation Receive functions. The Arbitration State Machine function enables the Transmit function to advertise and acknowledge the abilities. Upon indication of acknowledgment, the Arbitration State Machine function determines the highest common denominator using the **Priority Resolution** function and enables the appropriate technology-dependent PHY. The Arbitration State Machine is implemented in accordance with the *Figure 73-11* as specified in *Clause 73 of IEEE 802.3 standard*.

1.1.2.4 APB Target Registers Interface (Ask a Question)

A 32-bit APB target register interface is used to configure, initialize, and read the status of the Auto-Negotiation TX and Auto-Negotiation RX. The details of the Auto-Negotiation registers are specified in the 7.1.1.1. Auto-Negotiation Registers section of this document. Logic is implemented in the RTL to synchronize the Auto-Negotiation register bits to the APB Target Clock (PCLK).

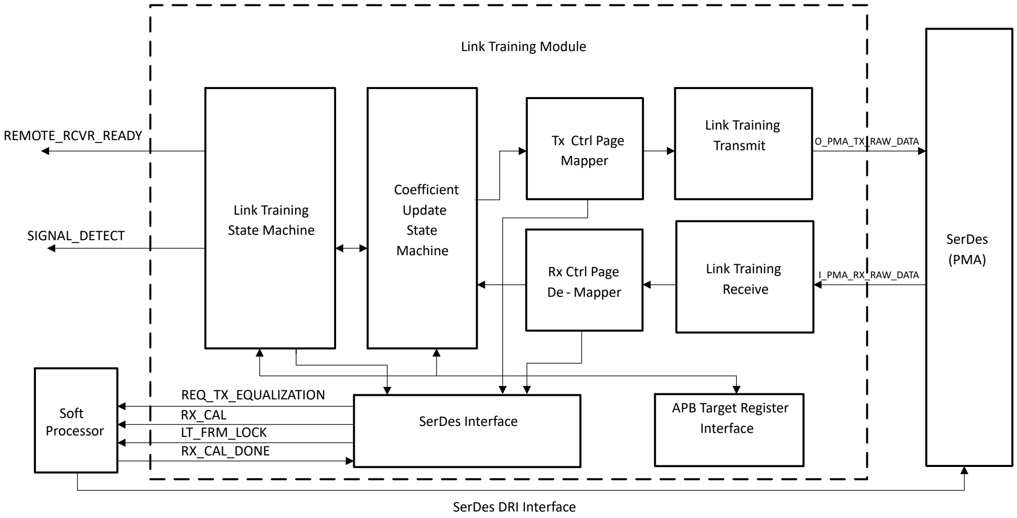
1.1.3 Link Training (Ask a Question)

The Link Training module is implemented in accordance with the *IEEE 802.3 Clause 72* standard. Link Training is initiated after Auto-Negotiation is completed and is done at a 10G line rate. Once the Auto-Negotiation Arbitration State Machine reaches the AN_GOOD_CHK state, an interrupt is raised to the processor to initiate Link Training between the local device and the Link Partner.

Upon completion of the training within `max_wait_timer` (500 ms), an interrupt is raised indicating successful completion of Link Training between the local device and the Link Partner. If the training fails to complete before the expiration of the `max_wait_timer`, the Link Training State Machine enters the training fail state, and the training is re-initiated by the embedded software driver. Upon unsuccessful completion of the Link Training, the `max_wait_timer` sequence falls back to Auto-Negotiation.

The following figure shows the top-level block diagram of the Link Training.

Figure 1-4. Link Training Block Diagram



The Link Training comprises of the following modules:

- Link Training Transmit
- Link Training Receive
- Coefficient Update State Machine
- Link Training State Machine
- Link Training TX Control Page Mapper
- Link Training RX Control Page De-Mapper
- SerDes Interface
- APB Target Register Interface

1.1.3.1 Link Training Transmit Module (Ask a Question)

The Link Training Transmit module is the PCS encoder module (DME Encoding). The Link Training frames are formed and encoded by this module, which are transmitted to the Link partner through the SerDes interface. For more information on the training frame structure, see the training frame structure and its fields explained in *section 72.6.10.2 of the IEEE 802.3* standard.

1.1.3.2 Link Training Receive Module (Ask a Question)

This is the PCS decoder module. The received training frames are decoded and the extracted control frames are sent to the De-Mapper module of the RX control page.

1.1.3.3 Coefficient Update State Machine (Ask a Question)

The Coefficient Update State Machine defines the process for updating transmit equalizer coefficients in response to the requests from the Link Partner and defines the coefficient update status to be reported in outgoing training frames. The Coefficient Update State Machine is implemented in accordance with the *Figure 72-6* as specified in *Clause 72 of IEEE 802.3 standard*.

1.1.3.4 Link Training State Machine (Ask a Question)

The Link Training State Machine is implemented in accordance with the *Figure 72.5 of the IEEE 802.3* standard.

1.1.3.5 Link Training TX Control Page Mapper (Ask a Question)

The Link Training TX Control Page Mapper module receives the coefficient and status fields from the Coefficient Update State Machine module. It maps the coefficient and the status fields into a 32-bit link training control page and then transmits it to the Link Training Transmit module, which further frames it into a training frame and transmits it to the SerDes interface.

1.1.3.6 Link Training RX Control Page De-Mapper (Ask a Question)

The Link Training RX Control Page De-Mapper module receives the 32-bit control page from the Link Training Receiver module. It extracts the coefficient update and status fields and sends the respective information to the Coefficient Update State Machine module.

1.1.3.7 APB Target Interface (Ask a Question)

A 32-bit APB target interface is used to configure, initialize, and read the status of the Link Training registers. The details of the Link Training registers are provided in the 7.1.1.2. Link Training Registers section. Logic is implemented in the RTL to synchronize the Link training register bits to the APB Target clock (PCLK).

1.1.4 FEC TX and RX (Ask a Question)

FEC Transmit and Receive modules are implemented as per the Clause 74, IEEE standard. The FEC sublayer can be placed in between the PCS and PMA sublayers of the 10GBASE-KR Physical Layer implementation. The 10GBASE-KR standard uses the FEC sublayer to increase the performance on a broader set of backplane channels.

The implemented FEC Module has the following features:

- FEC Encoder/Decoder
- Scrambler/Descrambler
- Optional Error Indication

2. IP Core Parameters and Interface Signals (Ask a Question)

This section discusses the parameters in the Core10GBaseKR_PHY Configurator settings and I/O signals.

2.1 Configuration Settings (Ask a Question)

The following table lists the Core10GBaseKR_PHY GUI parameters for configuring the core.

Important: The Name column in the following table shows the actual parameter names used in RTL. The Description column starts with the parameter names as they appear in the Core10GBaseKR_PHY GUI configurator.

Table 2-1. GUI Parameters

Name	Valid Values	Default Value	Description
Personality			
LT_AN_ENABLE	1 and 0	1	**10G Type** 1—Enable the AN and LT function 0—Disable the AN and LT function **Note:** When the value is '0'. AN and LT operations are disabled and the IP supports only the IEEE 802.3 – Clause 49 operation.
FCFEC_EN	0 and 1	0	**FCFEC_EN** 1—Enable the FireCode FEC Function 0—Disable the FireCode FEC Function

2.2 Inputs and Outputs Signals (Ask a Question)

The following table lists the Core10GBaseKR_PHY clock ports.

Table 2-2. Core10GBaseKR_PHY Clock Ports

Name	Width	Direction	Description
I_TX_CLK	1	Input	Transmit clock of 322.266 MHz provided by the transceiver
I_RX_CLK	1	Input	Recovered clock of 322.266 MHz provided by the transceiver
PCLK	1	Input	Initiator clock input
I_XGMII_TXCLK	1	Input	XGMII TX clock of 156.25 MHz
I_XGMII_RXCLK	1	Input	XGMII RX clock of 156.25 MHz
I_PCS73_TX_CLK	1	Input	PCS73 TX Gearbox Clock **Note:** Connects to the TX Lane clock of PF_XCVR
I_PCS73_RX_CLK	1	Input	PCS73 RX Gearbox Clock **Note:** Connects to the RX Lane clock of PF_XCVR

The following table lists the input and output ports of the Core10GBaseKR_PHY. The Clock Domain column in the table indicates the input ports and the output ports expected to be synchronized in the respective clock domain.

Table 2-3. Core10GBaseKR_PHY Input and Output Ports

Name	Clock Domain	Width	Direction	Description
SYSTEM RESETS				
I_TXCLK_RESETN	I_TX_CLK	1	Input	Active low asynchronous reset input to reset Transmit Logic. The reset input is synchronized internally to I_TX_CLK.
I_RXCLK_RESETN	I_RX_CLK	1	Input	Active low asynchronous reset input to reset Receive Logic. The reset input is synchronized internally to I_RX_CLK.

...........continued

Name	Clock Domain	Width	Direction	Description
I_PCS49_TX_SRESET	I_TX_CLK	1	Input	Active high asynchronous reset input to reset PCS Transmit Logic. The reset input is synchronized internally to I_TX_CLK.
I_PCS49_RX_SRESET	I_RX_CLK	1	Input	Active high asynchronous reset input to reset PCS Receive Logic. The reset input is synchronized internally to I_RX_CLK.
PRESETN	PCLK	1	Input	Active low asynchronous reset. The reset input is synchronized internally to PCLK.
I_XGMII_TXRSTN	I_XGMII_TXCLK	1	Input	Active low asynchronous reset input to reset Transmit Logic. The reset input is synchronized internally to I_XGMII_TXCLK.
I_XGMII_RXRSTN	I_XGMII_RXCLK	1	Input	Active low asynchronous reset input to reset Receive Logic. The reset input is synchronized internally to I_XGMII_RXCLK.
10GE TX PCS SIGNALS				
I_CFG_PCS49_TX_BYPASS_SCRAMBLER	I_TX_CLK	1	Input	When asserted high, the scrambler is bypassed. Normally tied to 1'b0
I_CFG_PCS49_TX_TEST_PRBS31_EN	I_TX_CLK	1	Input	When asserted high, the core is continuously sourcing PRBS31, as defined by Clause 49. This signal is the highest priority, that is, if asserted, the value on the other I_CFG_PCS49_* signal is ignored.
I_CFG_PCS49_TX_TEST_PATTERN_EN	I_TX_CLK	1	Input	When I_CFG_PCS49_TX_TEST_PATTERN_EN is asserted, the transmitter sources either a square pattern or a pseudo-random pattern as defined by Clause 49. The specific pattern is determined by I_CFG_PCS49_TX_TEST_PATTERN_TYPE_SEL.
I_CFG_PCS49_TX_TEST_PATTERN_TYPE_SEL	I_TX_CLK	1	Input	When asserted high, the core transmits a square pattern. The square pattern is sourced as 16'hf0f0 towards the PMA. When asserted low, the core sources a pseudo-random pattern as determined by I_CFG_PCS49_TX_TEST_PATTERN_DATA_SEL, I_CFG_PCS49_TX_TEST_PATTERN_SEED_A, and I_CFG_PCS49_TX_TEST_PATTERN_SEED_B.
I_CFG_PCS49_TX_TEST_PATTERN_DATA_SEL	I_TX_CLK	1	Input	When asserted high, the core uses 64 zeroes as the data-pattern. When asserted low, the core uses 64-bit encoding for two Local Fault ordered sets. See Clause 49 for clarification.
I_CFG_PCS49_TX_TEST_PATTERN_SEED_A	I_TX_CLK	58	Input	These values are used for the scrambler seed while running in pseudo-random test-mode. See Clause 49 for clarification.
I_CFG_PCS49_TX_TEST_PATTERN_SEED_B	I_TX_CLK	58	Input	These values are used for the scrambler seed while running in pseudo-random test-mode. See Clause 49 for clarification.
10GE RX PCS SIGNALS				
I_CFG_PCS49_RX_BYPASS_SCRAMBLER	I_RX_CLK	1	Input	When asserted high, the scrambler is bypassed. Normally tied to 1'b0
I_CFG_PCS49_RX_TEST_PRBS31_EN	I_RX_CLK	1	Input	When asserted high, the core receives PRBS31 input. This signal has highest priority, if asserted the signals on the other I_CFG_PCS49_RX_* are ignored.
I_CFG_PCS49_RX_TEST_PATTERN_EN	I_RX_CLK	1	Input	When I_CFG_PCS49_RX_TEST_PATTERN_EN is asserted, the receiver receives either a square pattern or a pseudo-random pattern as defined by Clause 49. The specific pattern received will be determined by I_CFG_PCS49_RX_TEST_PATTERN_TYPE_SEL.
I_CFG_PCS49_RX_TEST_PATTERN_TYPE_SEL	I_RX_CLK	1	Input	When asserted high, the core receives a square pattern. When asserted low, the core receives a pseudo-random pattern as determined by I_CFG_PCS49_RX_TEST_PATTERN_DATA_SEL
I_CFG_PCS49_RX_TEST_PATTERN_DATA_SEL	I_RX_CLK	1	Input	When asserted high, the core will receive 4 zero's as the data-pattern. When asserted low, the core will receive 64-bit encoding for two Local Fault ordered sets. See Clause 49 for clarification.
10GE PCS RX STATUS & STATS				
O_PCS49_RX_BLOCK_LOCK	I_RX_CLK	1	Output	The signal is asserted when the receiver acquires block delineation
O_PCS49_RX_HI_BER	I_RX_CLK	1	Output	The signal is asserted when the ber_cnt equals or exceeds 16 indicating a bit error ratio >10^{-4}
O_PCS49_RX_STATUS	I_RX_CLK	1	Output	This signal indicates that the receiver is in block lock and not in hi_ber state.
O_PCS49_RX_BER_STRB	I_RX_CLK	1	Output	The signal strobes every time 125 us_timer_done asserts, that is, every 125 µs per the Clause 49 specification. The O_PCS49_RX_BER_CNT signal is updated on the same event.
O_PCS49_RX_BER_CNT	I_RX_CLK	8	Output	An 8-bit counter that counts each time BER_BAD_SH state is entered. The counter reflects the number of events since the last time O_PCS49_RX_BER_STRB was asserted. The counter value is updated at the same time as O_PCS49_RX_BER_STRB, and remains stable until the next O_PCS49_RX_BER_STRB event. The maximum value of the signal is 16.
O_PCS49_RX_TEST_MODE_ERR_STRB	I_RX_CLK	1	Output	The signal strobes to indicate an update on O_PCS49_RX_TEST_MODE_ERR_CNT. **Note:** This signal asserts irrespective of errors being present.
O_PCS49_RX_TEST_MODE_ERR_CNT	I_RX_CLK	8	Output	The receive test pattern error counter. This counter is used to indicate PRBS31 errors and pseudo-random-sequence errors. When the receiver is not running in one of those two modes, the counter is always zero, and the associated strobe will never assert.
O_PCS49_RX_ERRORED_BLOCK_CNT_STRB	I_RX_CLK	1	Output	When the receiver is in normal mode, this signal strobes each time RX_E state is entered.
APB Interface				
PWRITE	PCLK	1	Input	APB write/read enable, active high
PADDR	PCLK	16	Input	APB address
PSEL	PCLK	1	Input	APB select
PENABLE	PCLK	1	Input	APB enable
PWDATA	PCLK	32	Input	APB data input
PRDATA	PCLK	32	Output	APB data output
PREADY	PCLK	1	Output	Ready The Target uses this signal to extend an APB transfer
PSLVERR	PCLK	1	Output	This signal indicates a transfer failure.

...........continued

Name	Clock Domain	Width	Direction	Description
XGMII Interface				
I_XGMII_RXD	I_XGMII_TXCLK	64	Input	XGMII RX Data
I_XGMII_RXC	I_XGMII_TXCLK	8	Input	XGMII RX Control
O_XGMII_TXD	I_XGMII_RXCLK	64	Output	XGMII TX Data
O_XGMII_TXC	I_XGMII_RXCLK	8	Output	XGMII TX Control
PMA				
O_PMA_TX_RAW_DATA	I_TX_CLK	32	Output	PMA TX Raw Data
I_PMA_RX_RAW_RDY	I_RX_CLK	1	Input	PMA RX Ready signal
I_PMA_RX_RAW_EN	I_RX_CLK	1	Input	PMA RX enable signal
I_PMA_RX_RAW_DATA	I_RX_CLK	32	Input	PMA RX Raw Data
PCS 73 Gear Box				
I_PCS73_DATA	I_PCS73_RX_CLK	32	Input	PCS73 RX Gearbox Data **Note:** Connects to the RX Lane data of PF_XCVR.
I_PCS73_VAL	I_PCS73_RX_CLK	1	Input	PCS73 RX Gearbox Valid **Note:** Connects to the RX Lane data of PF_XCVR.
AN Status Signals				
AN_LINK_GOOD_CHECK	I_PCS73_TX_CLK	1	Output	AN Link Good Check Interrupt This signal is asserted when AN Link Good Check is received. It indicates that the rates are negotiated between the Local device and the Link Partner and Link Training can be started.
AUTONEG_COMPLETE	I_PCS73_TX_CLK	1	Output	Autoneg_complete If asserted, it indicates that Auto-Negotiation is completed. This signal must be asserted after both AN and LT are completed.
LT Status Signals				
REQ_TX_EQUALIZATION	I_TX_CLK	1	Output	req_tx_equalization Interrupt If this signal is asserted, it indicates that an increment or decrement request from the Link Partner to update the TX amplitude/ de-emphasis settings.
RX_CAL	I_RX_CLK	1	Output	Rx_cal Interrupt If asserted, it indicates that the Link partner updated its TX amplitude/de-emphasis settings. The DFE calibration must be initiated on the new received data.
LT_FRM_LOCK	I_TX_CLK	1	Output	lt_frm_lock Interrupt If asserted, this signal indicates that the linking training frame lock is achieved and receiver adaptation process can be started.
SIGNAL_DETECT	I_TX_CLK	1	Output	Signal_detect Interrupt If asserted, it indicates that the signal detect is achieved and the Link Training is complete.
RX_CAL_DONE	I_RX_CLK	1	Input	This is input signal to the IP indicating that the RX calibration is completed.
REMOTE_RCVR_READY	I_RX_CLK	1	Output	Signal indicating that the Link Partner receiver is ready.
TRAINING_FAIL	I_TX_CLK	1	Output	Signal indicating that the Link Training is not completed and the 500 ms Timer has elapsed.
Elastic Buffer Status Signals				
ELASTIC_BUFFER_OVERFLOW	I_XGMII_RXCLK	1	Output	Elastic Buffer Overflow signal
ELASTIC_BUFFER_UNDERFLOW	I_XGMII_TXCLK	1	Output	Elastic Buffer Underflow signal
LOS Port				
XCVR_LOS	I_TX_CLK	1	Output	Loss of signal to the transceiver. Asserting this signal holds the transceiver in lock to the reference mode.
FEC Signals				
I_CFG_FCFEC_RX_ENABLE_ERROR_INDICATION	I_RX_CLK	1	Input	When asserted the core reports uncorrected errors on the header bits.
O_FCFEC_RX_BLOCK_LOCK	I_RX_CLK	1	output	The signal is asserted when the receiver acquires block delineation.
O_FCFEC_RX_BLOCK_EN	I_RX_CLK	1	output	After the O_FCFEC_RX_BLOCK_LOCK is asserted, this signal asserts at the end of every FEC block. It is used to indicate the reception and processing of one FEC block
O_FCFEC_RX_CORRECTED_BLOCK	I_RX_CLK	1	output	This signal is asserted when the core experienced a corrected block.
O_FCFEC_RX_UNCORRECTED_BLOCK	I_RX_CLK	1	output	This signal is asserted when the core experienced an uncorrected block.

3. Timing Diagrams (Ask a Question)

This section discusses various timing diagrams.

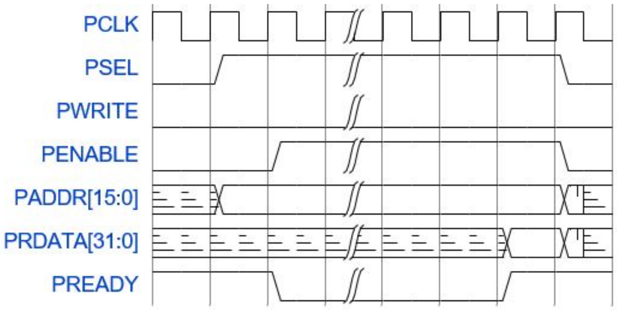
3.1 APB Interface (Ask a Question)

This section describes the read and write operation of APB interface.

3.1.1 APB Read Timing (Ask a Question)

The following figure shows the timing diagram for an APB read access.

Figure 3-1. APB Read Timing Diagram

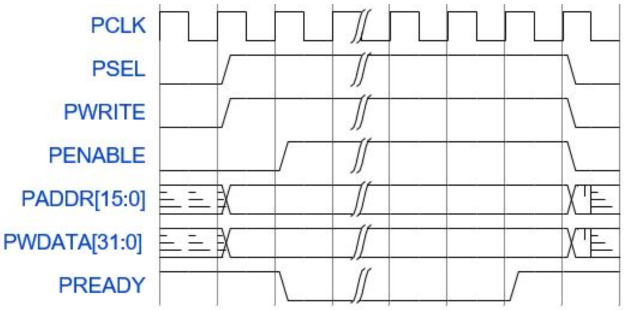


As per the preceding figure, the read transfer starts with PADDR, PWRITE, and PSEL changing after the rising edge of PCLK. The first clock cycle of the transfer is called the setup phase. After the following clock edge, the PENABLE is asserted, and PREADY is de-asserted; this indicates that the access phase is taking place. PADDR, PWRITE, PSEL, and PENABLE all remain valid throughout the access phase. The transfer completes at the end of the cycle where PREADY is asserted. During this cycle, PRDATA is valid. PENABLE is de-asserted at the end of the transfer. PSEL also goes low unless the transfer is followed immediately by another transfer to the same peripheral.

3.1.2 APB Write Timing (Ask a Question)

The following figure shows the timing diagram for an APB write access.

Figure 3-2. APB Write Timing Diagram



As per the preceding figure, the write transfer starts with PADDR, PWDATA, PWRITE, and PSEL changing after the rising edge of PCLK. The first clock cycle of the transfer is called the setup phase. After the following clock edge, the PENABLE is asserted, and PREADY is de-asserted; this indicates that the access phase is taking place. PADDR, PWDATA, PWRITE, PSEL, and PENABLE all remain valid throughout the access phase. The transfer completes at the end of the cycle where PREADY is asserted. PENABLE is de-asserted at the end of the transfer. PSEL also goes low unless the transfer is followed immediately by another transfer to the same peripheral.

4. Implementation of Core10GBaseKR_PHY in Libero Design Suite (Ask a Question)

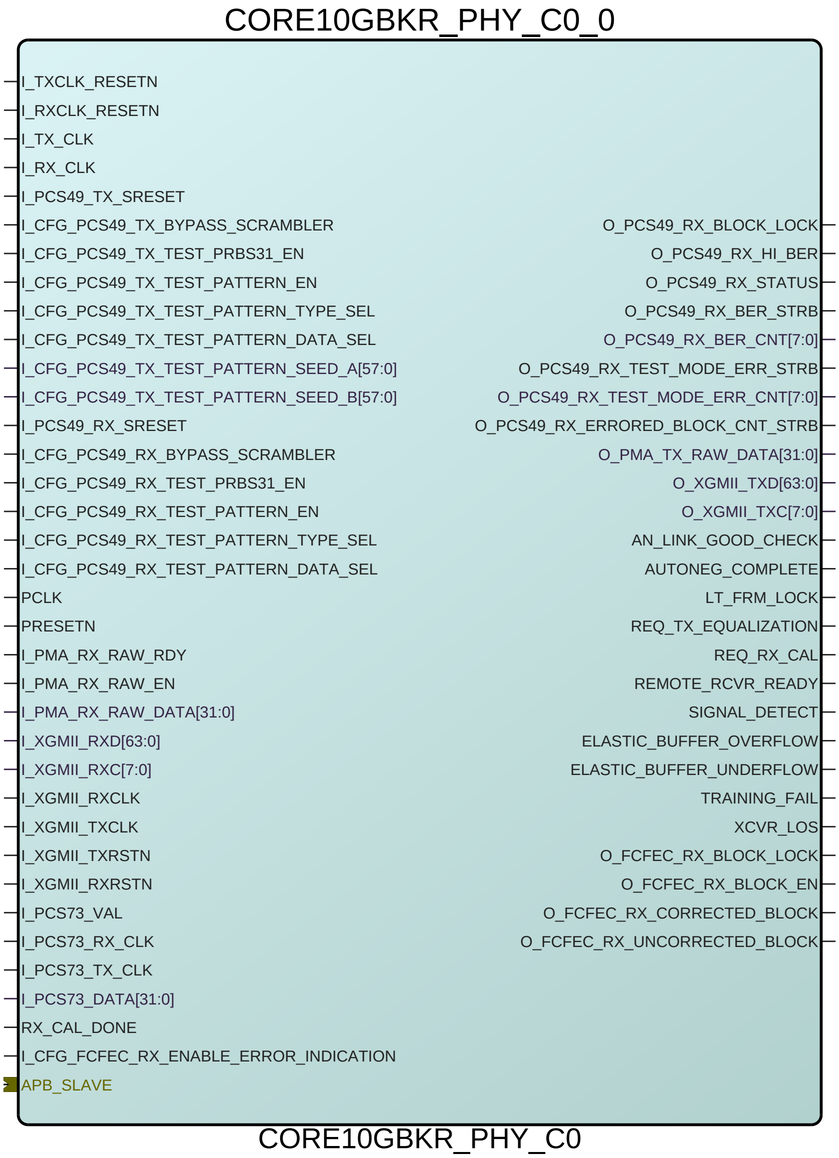
This section describes implementation of the Core10GBaseKR_PHY in Libero Design Suite.

4.1 Smart Design (Ask a Question)

Core10GBaseKR_PHY is available for download to the SmartDesign IP catalog through the Libero SoC web repository. To know how to create a Smart Design project, see SmartDesign User Guide .

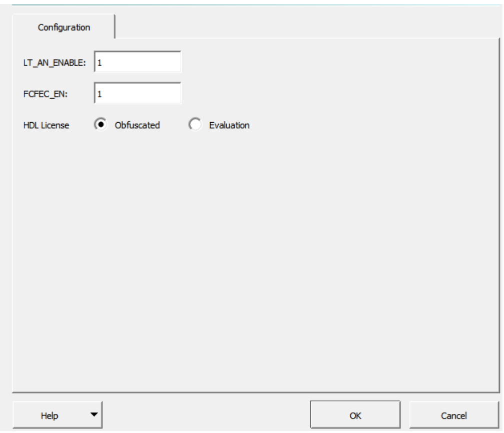
The following figure shows instantiated view of Core10GBaseKR_PHY on the SmartDesign canvas.

Figure 4-1. Core10GBaseKR_PHY Fill I/O View



The following figure shows the options available under the **Configuration** tab.

Figure 4-2. Core10GBaseKR_PHY SmartDesign Configuration GUI—Configuration



LT_AN_ENABLE: 1

FCFEC_EN: 1

HDL License ● Obfuscated ○ Evaluation

4.1.1 Simulation Flows (Ask a Question)

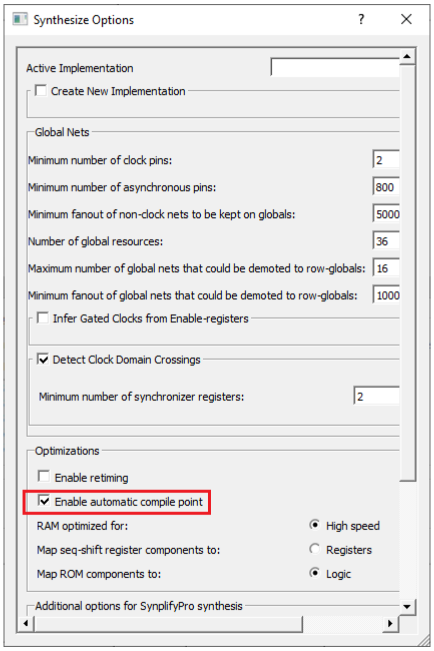
User testbench is not provided for Core10GBaseKR_PHY v3.0. However, the recommendation to setup the simulation testbench is provided in User testbench section.

4.1.2 Synthesis in Libero SoC (Ask a Question)

To run the synthesis with the configuration selected in the configuration GUI, set the design root as required. To meet the Core10GBaseKR_PHY timing requirements, the **Enable automatic compile point** option must be applied. Complete the following steps to apply a synthesis constraint:

1. In the **Design Flow** tab, under **Implement Design**, right-click **Synthesize**, and then click **Configure Options**.
2. In the **Synthesis Options** window, select the **Enable automatic compile point** option, as shown in the following figure.
3. Click **OK**.

Figure 4-3. Synthesis Constraint



Active Implementation

□ Create New Implementation

Global Nets

Minimum number of clock pins: 2

Minimum number of asynchronous pins: 800

Minimum fanout of non-clock nets to be kept on globals: 5000

Number of global resources: 36

Maximum number of global nets that could be demoted to row-globals: 16

Minimum fanout of global nets that could be demoted to row-globals: 1000

□ Infer Gated Clocks from Enable-registers

☑ Detect Clock Domain Crossings

Minimum number of synchronizer registers: 2

Optimizations

□ Enable retiming

☑ Enable automatic compile point

RAM optimized for: ● High speed

Map seq-shift register components to: ○ Registers

Map ROM components to: ● Logic

Additional options for SynplifyPro synthesis

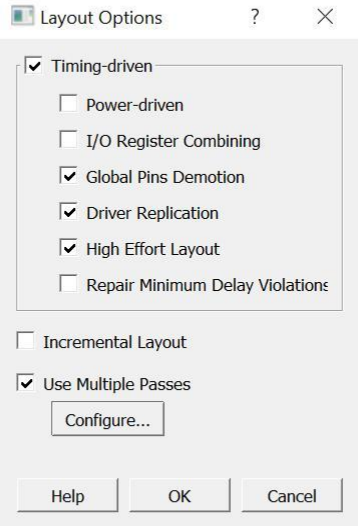
4. After setting the synthesis constraint, double-click **Synthesize** to run the synthesis.

4.1.3 Place-and-Route in Libero SoC (Ask a Question)

Place-and-Route constraints must be applied to meet the Core10GBaseKR_PHY timing requirement. Follow these steps to apply Place-and-Route constraints:

1. In the **Design Flow** tab, under **Implement Design**, right-click **Place and Route**, and click **Configure Options**.
2. In the **Layout Options** window, select the **Use Multiple Passes** option, and, click **Configure**, as shown in the following figure.

Figure 4-4. Place-and-Route Constraints



Layout Options

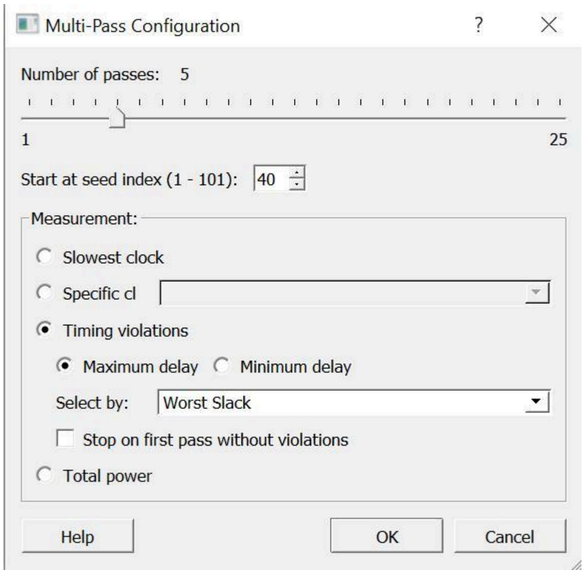
☑ Timing-driven
- ☐ Power-driven
- ☐ I/O Register Combining
- ☑ Global Pins Demotion
- ☑ Driver Replication
- ☑ High Effort Layout
- ☐ Repair Minimum Delay Violations

☐ Incremental Layout

☑ Use Multiple Passes

3. In the **Multi-Pass configuration** window, set **Number of passes** to **14**, and select the **Stop on first pass without violations** option, as shown in the following figure.

Figure 4-5. Place-and-Route Constraints Continue..



Multi-Pass Configuration

Number of passes: 5

Start at seed index (1 - 101): 40

Measurement:
- ○ Slowest clock
- ○ Specific cl
- ● Timing violations
  - ● Maximum delay ○ Minimum delay
  - Select by: Worst Slack
  - ☐ Stop on first pass without violations
- ○ Total power

4. Click **OK**.
5. After applying constraint, to run the place-and-route, double-click **Place-and-Route**.

4.1.4 System Integration (Ask a Question)

This section provides hints to ease the integration of Core10GBaseKR_PHY.

4.2 Verification (Ask a Question)

This section describes verification of the Core10GBaseKR_PHY.

4.2.1 Testbench (Ask a Question)

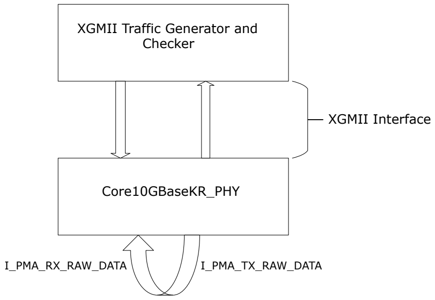
A unified testbench is used to verify and test Core10GBaseKR_PHY called as user testbench.

4.2.1.1 User Testbench (Ask a Question)

Case1: The testbench needed to verify the data path alone in the Loopback mode is shown in the Figure 4-6. The XGMII traffic generator and checker model is needed to generate and receive the XGMII traffic. The data Loopback needs to be done at the Line side (towards XCVR side). Device Under Test (DUT) encodes the data received from the XGMII model as per the Clause 49 and transmits to the Line side. The Looped back data is received by the IP and decoded as per Clause 49 and translated to the XGMII format.

The XGMII checker model needs to verify the data integrity of the received XGMII packet.

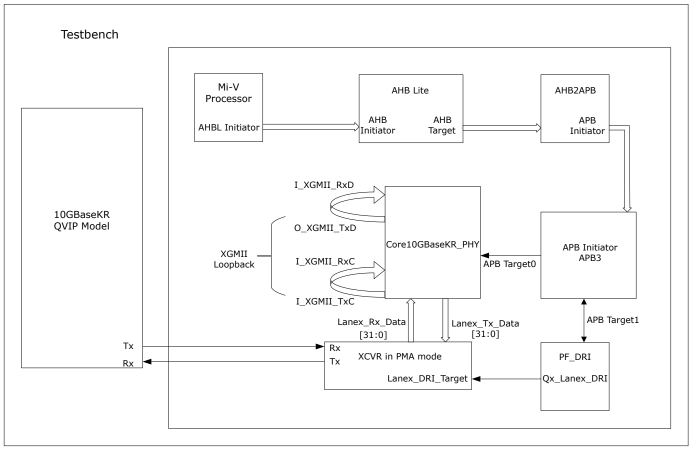
Figure 4-6. Simulation Setup to Verify Data Path



Case2: To carry out the simulation for Clause 73 (Auto-Negotiation), Clause 72 (Link Training), and the Clause 49 (data integrity) QVIP model for 10GBaseKR is needed.

The following figure shows the simulation setup needed to perform the simulation. DUT comprises of the system with Mi-V processor (that is, to run the embedded software needed for Auto-Negotiation and Link Training), AHB to APB Bridge, PF_DRI IP to access and configure the Transceiver registers, and APB target interface to configure the Core10GBaseKR_PHY. The data Loopback is done at the XGMII side.

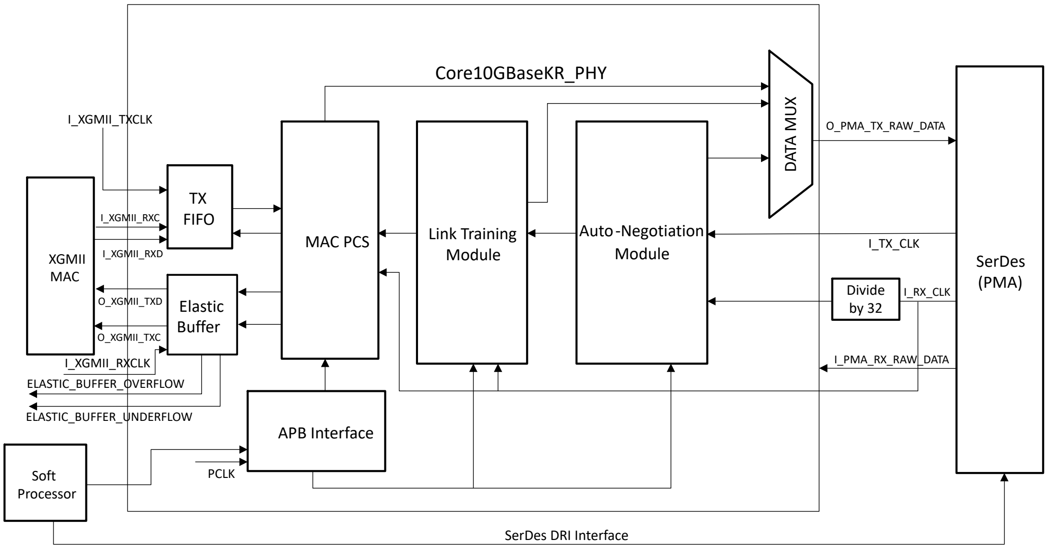
Figure 4-7. Simulation Using QVIP Model



5. Clocking and Resets (Ask a Question)

The following figure shows the clocking used in the Core10GBaseKR_PHY.

Figure 5-1. Core10GBaseKR_PHY—Clocking



5.1 MAC Side Clocks (Ask a Question)

The XGMII signals interfacing the MAC operate at the frequency of 156.25 MHz for the 64-bit data interface. The XGMII TX interface operates on I_XGMII_TXCLK and RX interface with I_XGMII_RXCLK.

The FIFO and the Elastic buffer in the design acts as the CDC interface.

5.1.1 Transceiver Side Clocks (Ask a Question)

The transceiver generates two clock outputs, I_TX_CLK and I_RX_CLK (recovered RX clock), at 322.265 MHz (32-bit interface). All the modules in the design on the right side of the TXFIFO and Elastic buffer operates with these clocks.

The TX logic operates with I_TX_CLK and RX logic with I_RX_CLK clock.

During the Auto-Negotiation operation, the transceiver generates the 9.765 MHz as the RX clock to the Auto-Negotiation module as the rate of operation is 312.5 Mbps.

The TX Logic in the Auto-Negotiation module still operates at 10 Gbps speed in the Oversampled Mode (322.265*32 = 10312.5 MHz).

5.1.2 APB Clock (Ask a Question)

The Register Interface and the SerDes DRI Interface operates with APB clock (PCLK) in the design.

6. Tool Flows (Ask a Question)

This section discusses tool-flow related information.

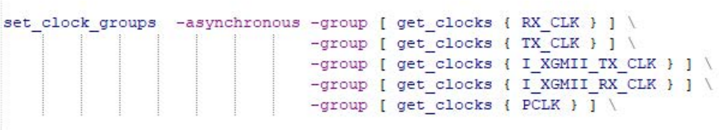
6.1 Timing Constraint (Ask a Question)

To meet the Core10GBaseKR_PHY timing requirement, it is important to provide a timing constraint. You must give a set clock group timing constraint to set the false path between asynchronous clocks.

The Core10GBaseKR_PHY uses the following clocks:

- I_XGMII_TX_CLK
- I_XGMII_RX_CLK
- TX_CLK
- RX_CLK
- PCLK

For example, if all the clocks mentioned above are asynchronous, then you can provide timing constraint as per the following example:



```
set_clock_groups  -asynchronous -group [ get_clocks { RX_CLK } ] \
                                -group [ get_clocks { TX_CLK } ] \
                                -group [ get_clocks { I_XGMII_TX_CLK } ] \
                                -group [ get_clocks { I_XGMII_RX_CLK } ] \
                                -group [ get_clocks { PCLK } ] \
```

The source clock name used in the preceding example is for reference only. The source name may differ in your design, and you must change the clock name in the constraint accordingly.

False path and multi-cycle path constraints that are used in the design for this IP are the following:

```
create_clock -name {JTAG_CLK} -period 250 -waveform {0 125 } [ get_ports { TCK } ]
set_false_path -from [ get_clocks { JTAG_CLK } ] -to [ get_clocks { BASEKR_CMN/SYS_FAB_CCC/
PF_CCC_C0_0/pll_inst_0/OUT1 } ]
set_false_path -from [ get_clocks { BASEKR_CMN/SYS_FAB_CCC/PF_CCC_C0_0/pll_inst_0/OUT1 } ] -
to [ get_clocks { JTAG_CLK } ]
set_false_path -from [ get_pins { BASEKR_IP_0_inst_0/XCVR_ERM/
I_XCVR/LANE0/RX_FWF_CLK } ] -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/rstSync_I_PCS49_RX_SRESET/tbi_txclk_irst1/ALn
BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/rstSync_I_PCS49_RX_SRESET/
tbi_txclk_irst2/ALn BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/
rstSync_I_RXCLK_RESETN/tbi_txclk_irst1/ALn BASEKR_IP_0_inst_0/CORE10GBKR_PHY/
CORE10GBKR_PHY_C0_0/rstSync_I_RXCLK_RESETN/tbi_txclk_irst2/ALn BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/rstSync_I_TXCLK_RESETN/tbi_txclk_irst1/ALn
BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/rstSync_PRESETN/tbi_txclk_irst2/ALn } ]
set_false_path -through [ get_pins { BASEKR_IP_0_inst_0/MIV_RV32_BKR_IP/MIV_RV32_C0_0_0/
u_opsrv_0/gen_opsrv_debug.u_opsrv_debug_unit_0/* } ]

set_multicycle_path -setup 32 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/data_repeater_0/out_data/D } ]
set_multicycle_path -hold 32 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/data_repeater_0/out_data/D } ]

set_multicycle_path -setup_only 32 -through [ get_nets {BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/* } ] -to
[ get_clocks { BASEKR_IP_0_inst_0/XCVR_ERM/I_XCVR/LANE0/TX_CLK_R } ]
set_multicycle_path -hold 32 -through [ get_nets { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/* } ] -to
[ get_clocks { BASEKR_IP_0_inst_0/XCVR_ERM/I_XCVR/LANE0/TX_CLK_R } ]

set_clock_groups  -asynchronous -group [ get_clocks { BASEKR_IP_0_inst_0/
XCVR_ERM/I_XCVR/LANE0/RX_CLK_R } ]
set_clock_groups  -asynchronous -group [ get_clocks { BASEKR_IP_0_inst_0/
XCVR_ERM/I_XCVR/LANE0/TX_CLK_R } ]
set_clock_groups  -asynchronous -group [ get_clocks { BASEKR_CMN/SYS_FAB_CCC/
PF_CCC_C0_0/pll_inst_0/OUT0 } ]
set_clock_groups  -asynchronous -group [ get_clocks { BASEKR_CMN/SYS_FAB_CCC/
PF_CCC_C0_0/pll_inst_0/OUT1 } ]
```

```
set_multicycle_path -setup_only -start 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
m_erx_gen.m_erx/Xgmii_pcs49_EB_0/o_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -hold 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
m_erx_gen.m_erx/Xgmii_pcs49_EB_0/o_fifo_64/BUF_FIFO_0/* } ]


set_multicycle_path -setup 2 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]
set_multicycle_path -hold 2 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]
set_multicycle_path -setup 2 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]
set_multicycle_path -hold 2 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]
set_multicycle_path -setup 2 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]
set_multicycle_path -hold 2 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_erx_gen.m_erx/Xgmii_pcs49_EB_0/* } ]

set_multicycle_path -setup_only -start 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -hold 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]

set_multicycle_path -setup 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -hold 2 -from
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -setup 2 -through
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -hold 2 -through
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -setup 2 -to
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]
set_multicycle_path -hold 2 -to
[ get_pins { BASEKR_IP_0_inst_0/CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/
pcsc49_gen.Xgmii_pcsc49_txbuf_0/i_fifo_64/BUF_FIFO_0/* } ]

set_multicycle_path -setup 32 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/* } ]
set_multicycle_path -hold 32 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/*} ]
set_multicycle_path -setup 32 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/* } ]
set_multicycle_path -hold 32 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/*} ]
set_multicycle_path -setup 32 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/* } ]
set_multicycle_path -hold 32 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/shim_layer.shim_layer_0/*} ]


set_multicycle_path -setup 32 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
set_multicycle_path -hold 32 -from [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
set_multicycle_path -setup 32 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
set_multicycle_path -hold 32 -through [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
set_multicycle_path -setup 32 -to [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
set_multicycle_path -hold 32 -to  [ get_pins { BASEKR_IP_0_inst_0/
CORE10GBKR_PHY/CORE10GBKR_PHY_C0_0/ms_10gBaseKR_top/m_c73_link_gen.m_c73_link/m_rx/* } ]
```

7. Register Map and Descriptions (Ask a Question)

7.1 Register Summary (Ask a Question)

This section provides the overview of APB registers, Clocking and resets, Transceiver side clocks, and APB clock.

7.1.1 APB Register Map (Ask a Question)

The following table lists the APB register maps description.

Table 7-1. Register Map

Address PADDR [11:8]	Name
0x0	Auto-Negotiation
0x4	Link Training
0x8	Transmit Control
0x9	Receive Status

The following tables list the APB functionality. The offset address is specified by PADDR [7:2].

Important: The following nomenclature are used:

- R—Read only
- R/W—Read Write
- R/Wc—Read Write, Self-clearing
- Rc—Clear on Read
- W—Write only

7.1.1.1 Auto-Negotiation Registers (Ask a Question)

The following table lists the Auto-Negotiation register description.

Table 7-2. Auto-Negotiation Registers

Address Offset PADDR [7:2]	Register [15:0]
0x0	AN Control Register
0x1	AN Status Register
0x10	Mr_adv_ability [16:1]
0x11	Mr_adv_ability [32:17]
0x12	Mr_adv_ability [48:33]
0x13	Mr_lp_base_page_ability[16:1]
0x14	Mr_lp_base_page_ability[32:17]
0x15	Mr_lp_base_page_ability[48:33]
0x16	Mr_xnp_transmit[16:1]
0x17	Mr_xnp_transmit[32:17]
0x18	Mr_xnp_transmit[48:33]
0x19	Mr_lp_xnp_ability[16:1]
0x1a	Mr_lp_xnp_ability[32:17]
0x1b	Mr_lp_xnp_ability[16:1]

7.1.1.1.1 Auto-Negotiation Control Register—(0x0) (Ask a Question)

The following table lists the Auto-Negotiation control register description.

Table 7-3. Auto-Negotiation Control Register

Bit	Type	Function	Default	Description
15	R/Wc	AN_RESET	0x0	Main_reset. Self clears. When reads as 0, the reset is complete.
14:13	R/W	Reserved	0x0	Reserved
12	R/W	AN_Enable	0x1	Auto-Negotiation Enable. Writing “1” in this register enables Auto-Negotiation
11:10	R/W	Reserved	0x0	Reserved
9	R/Wc	AN_RESTART	0x0	Restart_negotiation. Self clears, and always reads as 0.
8:0	R/W	Reserved	0x0	Reserved

7.1.1.1.2 Auto-Negotiation Status Register—(0x1) (Ask a Question)

The following table lists the Auto-Negotiation status register description.

Table 7-4. Auto-Negotiation status Register

Bit	Type	Function	Default	Description

15:12	R	an_state	0x0	This register returns the state variables of the Auto-Negotiation
				ST_AUTO_NEG_ENABLE = 0x0
				ST_TRANSMIT_DISABLE = 0x1
				ST_ABILITY_DETECT = 0x2
				ST_ACKNOWLEDGE_DETECT = 0x3
				ST_COMPLETE_ACKNOWLE = 0x4
				ST_AN_GOOD_CHECK = 0x5
				ST_AN_GOOD = 0x6
				ST_NEXT_PAGE_WAIT = 0x7
				ST_NEXT_PAGE_WAIT_TX_IDLE = 0x8
				ST_LINK_STATUS_CHECK =0x9
				ST_PARALLEL_DETECTION_FAULT = 0xA

11:10	R/W	Reserved	0x0	Reserved
9	Rc	Parallel_detect	0x1	Parallel_Detect_fault. Clear on read
8:7	R/W	Reserved	0x0	Reserved
6	Rc	Page_RX	0x0	Page_RX. If “1”, that Page is received from the link partner. Clear on read
5	R	AN_COMPLETE	0x0	If “1”, indicates that AN is complete
4	R/W	Reserved	0x0	Reserved
3	R	ANEG_Support	0x1	If “1”, indicates that Auto-Negotiation is supported
2:1	R/W	Reserved	0x0	Reserved
0	R	Autoneg_able	0x0	If “1”, indicates that Link Partner is Auto-Negotiation Able

7.1.1.1.3 Advertisement Ability Register-1—(0x10) (Ask a Question)

The following table lists the advertisement ability register-1 description.

Table 7-5. Advertisement Ability Register-1

Bit	Type	Function	Description
15:0	R/W	ADV_ABILITY_1	This register gives the value of the bits 16:1 of the Advertisement Ability Register. Bits [4:0]—Selector Field (S [4:0]) is a five-bit wide field, encoding 32 possible messages. Selector Field encoding definitions are shown in *Annex 28A of the IEEE 802.3* standard. Bits [9:5]—Echoed Nonce Field (E [4:0]) is a 5-bit wide field containing the nonce received from the link partner. Bits [12:10]—Pause Encoding as defined in *Annexure 28B of the IEEE 802.3* standard. Bit 13—Remote Fault (RF) of the base link codeword. The default value is logical zero. Bit 14—Acknowledge (Ack) is used by the Auto-Negotiation function to indicate that a device has successfully received its link partner's link codeword. Bit 15—Next Page (NP) bit. Support of Next Pages is mandatory. If the device does not have any Next Pages to send, the NP bit must be set to a logical zero.

7.1.1.1.4 Advertisement Ability Register-2—(0x11) (Ask a Question)

The following table lists the advertisement ability register-2 description.

Table 7-6. Advertisement Ability Register-2

Bit	Type	Function	Description
31:16	R/W	ADV_ABILITY_2	This register gives the value of the bits 32:17 of the Advertisement Ability Register. Bits[20:16]—Transmitted Nonce Field Bits[31:21]—Technology Ability Field bits [10:0]. Technology Ability Field is a 25-bit wide-field containing information indicating supported technologies specific to the selector field value when used with the Auto-Negotiation for Backplane Ethernet.

7.1.1.1.5 Advertisement Ability Register-3—(0x12) (Ask a Question)

The following table lists the advertisement ability register-3 description.

Table 7-7. Advertisement Ability Register-3

Bit	Type	Function	Description
47:32	R/W	ADV_ABILITY_3	This register gives the value of the bits 48:33 of the Advertisement Ability Register. Bits [45:32]—Technology Ability Field bits [25:11] For Technology Ability Field encoding refer to *Table 73-4 of the IEEE* standard. Bits [47:46]—FEC Capability (FEC not supported in the current implementation). Bit 46—FEC ability, if set to '1', PHY has FEC Ability. Bit 47—FEC requested When the FEC requested bit is set to logical one, it indicates a request to enable FEC on the link.

7.1.1.1.6 LP Base Page Ability Register-1—(0x13) (Ask a Question)

The following table lists the link partner base page ability register-1 description.

Table 7-8. Link Partner Base Page Ability Register -1

Bit	Type	Function	Description
15:0	R	LP_BASE_PG_ABILITY_1	This register gives the value of the bits 16:1 of the Link Partner Base Page Ability Register. The bits definition are same as the Advertisement Ability Register-1.

7.1.1.1.7 LP Base Page Ability Register-2—(0x14) (Ask a Question)

The following table lists the link partner base page ability register-2 description.

Table 7-9. Link Partner Base Page Ability Register -2

Bit	Type	Function	Description
15:0	R	LP_BASE_PG_ABILITY_2	This register gives the value of the bits 32:17 of the Link Partner Base Page Ability Register. The bits definition are same as the Advertisement Ability Register-2.

7.1.1.1.8 LP Base Page Ability Register-3—(0x15) (Ask a Question)

The following table lists the link partner base page ability register-3 description.

Table 7-10. Link Partner Base Page Ability Register-3

Bit	Type	Function	Description
15:0	R	LP_BASE_PG_ABILITY_3	This register gives the value of the bits 48:33 of the Link Partner Base Page Ability Register. The bits definition are same as the Advertisement Ability Register-3.

7.1.1.1.9 XNP Transmit Register-1—(0x16) (Ask a Question)

The following table lists the next page transmit register-1 description.

Table 7-11. Next Page Transmit Register-1

Bit	Type	Function	Description
15:0	R	XNP_TRANSMIT_1	This register gives the value of the bits 16:1 of the XNP (Next Page) Transmit Register. Bit [10:0]—Message Code Field (M [10:0]) is an eleven-bit wide field, encoding 2048 possible messages. Message Code Field definitions are shown in *Annex 28C of the IEEE 802.3 specification*. Bit 11—Toggle (T) is used by the Arbitration function to ensure synchronization with the Link Partner during Next Page exchange. This bit shall always take the opposite value of the Toggle bit in the previously exchanged link codeword. Bit 12—Acknowledge 2 (Ack2) is used by the Next Page function to indicate that a device has the ability to comply with the message. Following are the settings of Ack2: 0—Cannot comply with message 1—Complies with message Bit 13—Message Page bit 0—Unformatted Page 1—Message Page Bit 14—Acknowledge. Acknowledge (Ack) is used by the Auto-Negotiation function to indicate that a device has successfully received its Link Partner's link codeword. Bit 15—Next Page bit 0—Last Page 1—Additional Next Page(s) will follow.

7.1.1.1.10 XNP Transmit Register-2—(0x17) (Ask a Question)

The following table lists the next page transmit register-2 description.

Table 7-12. Next Page Transmit Register-2

Bit	Type	Function	Description
15:0	R	XNP_TRANSMIT_2	This register gives the value of the bits [32:17] of the XNP Transmit Register. This value corresponds to the Bits [31:16] of the Unformatted Code Field of the Next Page Register.

7.1.1.1.11 XNP Transmit Register-3—(0x18) (Ask a Question)

The following table lists the next page transmit register-3 description.

Table 7-13. Next Page Transmit Register-3

Bit	Type	Function	Description
15:0	R	XNP_TRANSMIT_3	This register gives the value of the bits [48:33] of the XNP Transmit Register. This value corresponds to the Bits [47:32] of the Unformatted Code Field of the Next Page Register.

7.1.1.1.12 LP XNP Ability Register-1—(0x19) (Ask a Question)

The following table lists the link partner next page ability register-1 description.

Table 7-14. Link Partner Next Page Ability Register-1

Bit	Type	Function	Description
15:0	R	LP_XNP_ABILITY_1	This register gives the value of the bits [16:1] of the Link Partner Next Page Ability register. The bits definition are same as the XNP Transmit Register-1.

7.1.1.1.13 LP XNP Ability Register-2—(0x1a) (Ask a Question)

The following table lists the link partner next page ability register-2 description.

Table 7-15. Link Partner Next Page Ability Register-2

Bit	Type	Function	Description
15:0	R	LP_XNP_ABILITY_2	This register gives the value of the bits [32:17] of the Link Partner Next Page Ability Register. The bits definition are same as the XNP Transmit Register-2.

7.1.1.1.14 LP XNP Ability Register-3—(0x1b) (Ask a Question)

The following table lists the link partner next page ability register-3 description.

Table 7-16. Link Partner Next Page Ability Register-3

Bit	Type	Function	Description
15:0	R	LP_XNP_ABILITY_3	This register gives the value of the bits [48:33] of the Link Partner Next Page Ability Register. The bits definition are same as the XNP Transmit Register-3.

7.1.1.2 Link Training Registers (Ask a Question)

The following table lists the link training registers description.

Table 7-17. Link Training Registers

Address Offset PADDR [7:2]	Register [15:0]
0x0	Link Training Control Register
0x1	MAX WAIT TIMER Configuration Register
0x2	Frame TIMER Configuration Register
0x3	Preset MAIN-TAP Configuration Register
0x4	Preset POST-TAP Configuration Register
0x5	Preset PRE-TAP Configuration Register
0x6	Initialize MAIN-TAP Configuration Register
0x7	Initialize POST-TAP Configuration Register
0x8	Initialize PRETap Configuration Register
0x9	Maximum Coefficient Limit of MAIN-TAP Configuration Register
0xA	Minimum Coefficient Limit of MAIN-TAP Configuration Register
0xB	Maximum Coefficient Limit of POST-TAP Configuration Register
0xC	Minimum Coefficient Limit of POST-TAP Configuration Register
0XD	Maximum Coefficient Limit of PRE-TAP Configuration Register
0XE	Minimum Coefficient Limit of PRE-TAP Configuration Register
0xF	TX Equalization Register
0x10	Local Receiver Lock Register
0x11	TX New MAIN-TAP Register
0x12	TX New POST-TAP Register
0x13	TX New PRE-TAP Register
0x14	Training State Machine Status Register
0x15	TX Updated Status Register
0x16	Received Coefficient Status Register
0x18	TX Coefficient Configuration Register

...........continued

Address Offset PADDR [7:2]	Register [15:0]
0x1F	PRBS Error Word count Register
0x26	10GBaseKR Status Register

7.1.1.2.1 Link Training Control Register—(0x0) (Ask a Question)

The following table lists the link training control register description.

Table 7-18. Link Training Control Register

Bit	Type	Function	Default	Description
31:2	R/W	Reserved	0x0	Reserved
3	R/W	Initialize	0x0	If “1”, initialize coefficient is transmitted to the Link Partner.
2	R/W	PRESET	0x0	If “1”, preset coefficient is transmitted to the Link Partner.
1	R/W	RESTART_TRAINING	0x0	A write of “1”, to this field brings the link training IP into reset mode. To bring the Link Training IP out of reset, write a value of 0x0 into this register.
0	R/W	TRAINING_ENABLE	0x0	If “1”, enables the Link Training. This bit must be enabled from the software after the AN_GOOD_CHK interrupt is received by the embedded software.

7.1.1.2.2 Max Wait Timer Configuration Register—(0x1) (Ask a Question)

The following table lists the Max Wait Timer configuration register description.

Table 7-19. Max Wait Timer Configuration Register

Bit	Type	Function	Default	Description
31:0	R/W	MAX_WAIT_TIMER	0x99D1852	You must configure the value in the Max Wait Timer to 500 ms, the training state diagram enters into the training failure state once the timer reaches 500 ms.

7.1.1.2.3 Frame Wait Timer Configuration Register—(0x2) (Ask a Question)

The following table lists the Frame Wait Timer configuration register description.

Table 7-20. Frame Wait Timer Configuration Register

Bit	Type	Function	Default	Description
31:0	R/W	FRM_WAIT_TIMER	0x66	The value in the Frame Wait Timer register must be between 100 and 300 frames. The default value is configured to 100 frames.

7.1.1.2.4 Preset Main-Tap Configuration Register—(0x3) (Ask a Question)

The following table lists the Preset Main-Tap configuration register description.

Table 7-21. Preset Main-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	PRESET_MAIN_TAP	0x29	The value in this register gives the Main-Tap value, when you receive the preset coefficient from the Link Partner. The Link Training algorithm starts from this TAP value.

7.1.1.2.5 Preset Post-Tap Configuration Register—(0x4) (Ask a Question)

The following table lists the Preset Post-Tap configuration register description.

Table 7-22. Preset Post-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	PRESET_POST_TAP	0x10	The value in this register gives the Post-Tap value, when you receive the preset coefficient from the Link Partner. The Link Training algorithm starts from this TAP value.

7.1.1.2.6 Preset Pre-Tap Configuration Register—(0x5) (Ask a Question)

The following table lists the Preset Pre-Tap configuration register description.

Table 7-23. Preset Pre-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	PRESET_PRE_TAP	0x5	The value in this register gives the Pre-Tap, when you receive the preset coefficient from the Link Partner. The Link Training algorithm starts from this TAP value.

7.1.1.2.7 Initialize Main-Tap Configuration Register—(0x6) (Ask a Question)

The following table lists the Initialize Main-Tap configuration register description.

Table 7-24. Initialize Main-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	INIT_MAIN_TAP	0x1b	The value in this register gives the Main-Tap value, when you receive the initialize coefficient from the Link Partner.

7.1.1.2.8 Initialize Post-Tap Configuration Register—(0x7) (Ask a Question)

The following table lists the Initialize Post-Tap configuration register description.

Table 7-25. Initialize Post-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	INIT_POST_TAP	0x5	The value in this register gives the Post-Tap value, when you receive the initialize coefficient from the Link Partner.

7.1.1.2.9 Initialize Pre-Tap Configuration Register—(0x8) (Ask a Question)

The following table lists the Initialize Pre-Tap configuration register description.

Table 7-26. Initialize Pre-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	INIT_PRE_TAP	0x3	The value in this register gives the Pre-Tap, when you receive the initialize coefficient from the Link Partner.

7.1.1.2.10 Maximum Coefficient (Max Coeff) Limit of Main-Tap Configuration Register—(0x9) (Ask a Question)

The following table lists the Max Coeff limit of Main-Tap configuration register description.

Table 7-27. Max Coeff Main-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MAX_MAIN_TAP	0x29	The value in this register gives the maximum value of the Main-Tap setting, which is tested for the TX equalization for the optimum Main-Tap settings.

7.1.1.2.11 Minimum Coefficient (Min Coeff) Limit of Main-Tap Configuration Register—(0xA) (Ask a Question)

The following table lists the Min Coeff limit of Main-Tap configuration register description.

Table 7-28. Min Coeff Main-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MIN_MAIN_TAP	0x1A	The value in this register gives the minimum value of the Main-Tap setting, which is tested for the TX equalization for the optimum Main-Tap settings.

7.1.1.2.12 Max Coeff Limit of Post-Tap Configuration Register—(0xB) (Ask a Question)

The following table lists the Max Coeff limit of Post-Tap configuration register description.

Table 7-29. Max Coeff Post-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MAX_POST_TAP	0x10	The value in this register gives the maximum value of the Post-Tap setting, which is tested for the TX equalization for the optimum Post-Tap settings.

7.1.1.2.13 Min Coeff Limit of Post-Tap Configuration Register—(0xC) (Ask a Question)

The following table lists the Min Coeff limit of Post-Tap configuration register description.

Table 7-30. Min Coeff Post-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MIN_POST_TAP	0x0	The value in this register gives the minimum value of the Post-Tap setting, which is tested for the TX equalization for the optimum Post-Tap settings.

7.1.1.2.14 Max Coeff Limit of Pre-Tap Configuration Register—(0xD) (Ask a Question)

The following table lists the Max Coeff limit of Pre-Tap configuration register description.

Table 7-31. Max Coeff Pre-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MAX_PRE_TAP	0x5	The value in this register gives the maximum value of the Pre-Tap setting, which is tested for the TX equalization for the optimum Pre-Tap settings.

7.1.1.2.15 Min Coeff Limit of Pre-Tap Configuration Register—(0xE) (Ask a Question)

The following table lists the Min Coeff limit of pre-tap configuration register description.

Table 7-32. Min Coeff Pre-Tap Configuration Register

Bit	Type	Function	Default	Description
31:16	R/W	Reserved	0x0	Reserved
15:0	R/W	MIN_PRE_TAP	0x0	The value in this register gives the minimum value of the Pre-Tap setting, which is tested for the TX equalization for the optimum Pre-Tap settings.

7.1.1.2.16 TX Equalization Register—(0xF) (Ask a Question)

The following table lists the TX Equalization register description.

Table 7-33. TX Equalization Register

Bit	Type	Function	Default	Description
31:4	R/W	Reserved	0x0	Reserved
3	W	TX_EQUALIZATION_PRE_DONE	0x0	A write to “1”, in this field indicates that the tx_equalization is done for the Pre-Tap.
2	W	TX_EQUALIZATION_POST_DONE	0x0	A write to “1”, in this field indicates that the tx_equalization is done for the Post-Tap.
1	W	TX_EQUALIZATION_MAIN_DONE	0x0	A write to “1”, in this field indicates that the tx_equalization is done for the Main-Tap.
0	W	TX_EQUALIZATION_DONE	0x0	A write to “1”, in this field indicates that the tx_equalization is done for all the TAPs (Pre, Post, and Main).

7.1.1.2.17 Local Receiver Lock Register—(0x10) (Ask a Question)

The following table lists the local receiver lock register description.

Table 7-34. Local Receiver Status Register

Bit	Type	Function	Default	Description
31:1	R/W	Reserved	0x0	Reserved
0	R/W	local_rcvr_locked	0x0	Write "1", to this register from the software once the local receiver is ready.

7.1.1.2.18 TX New Main-Tap Register—(0x11) (Ask a Question)

The following table lists the TX new Main-Tap register description.

Table 7-35. TX Main-Tap Update Register

Bit	Type	Function	Default	Description
31:8	R/W	Reserved	0x0	Reserved
7:0	R	TX_NEW_MAIN_TAP	0x0	A value in this register gives the new Main-Tap value, used to update the TX equalization SerDes registers.

7.1.1.2.19 TX New Post-Tap Register—(0x12) (Ask a Question)

The following table lists the TX New Post-Tap register description.

Table 7-36. TX Post Tap Update Register

Bit	Type	Function	Default	Description
31:8	R/W	Reserved	0x0	Reserved
7:0	R	TX_NEW_Post_TAP	0x0	A value in this register gives the new Post-Tap value, used to update the TX equalization SerDes registers.

7.1.1.2.20 TX New Pre-Tap Register—(0x13) (Ask a Question)

The following table lists the TX New Pre-Tap register description.

Table 7-37. TX Pre-Tap Update Register

Bit	Type	Function	Default	Description
31:8	R/W	Reserved	0x0	Reserved
7:0	R	TX_NEW_Pre_TAP	0x0	A value in this register gives the new Pre-Tap value, used to update the TX equalization SerDes registers.

7.1.1.2.21 Training State Machine Status Register—(0x14) (Ask a Question)

The following table lists the training state machine status register description.

Table 7-38. LT State Machine Status Register

Bit	Type	Function	Default	Description
31:5	R/W	Reserved	0x0	Reserved
4	R	Remote_trained	0x0	If "1", it indicates that the remote receiver is ready.
3	R	Link_up	0x0	If "1", it indicates that the link is ready.

2:0	R	TRAINING_SM	0x0	The value in this register gives the current values of the Training State Machine.
				IDLE = 3'b000;
				INITIALIZE = 3'b001;
				SEND_TRAINING = 3'b011;
				TRAIN_LOCAL = 3'b010;
				TRAIN_REMOTE = 3'b110;
				LINK_READY = 3'b111;
				SEND_DATA = 3'b101;
				Parameter TRAINING_FAILURE = 3'b100;

7.1.1.2.22 TX Updated Status Register—(0x15) (Ask a Question)

The following table lists the TX updated status register description.

Table 7-39. Transmit Status Register

Bit	Type	Function	Default	Description
31:18	R/W	Reserved	0x0	Reserved
17:16	R	TX_updated_Status_pre	0x0	A value in this register gives the current status transmitted to the Link Partner for the Pre-Tap. 1 1 = maximum 1 0 = minimum 0 1 = updated 0 0 = not_updated
15:14	R	TX_updated_Status_post	0x0	A value in this register gives the current status transmitted to the Link Partner for the Post-Tap. 1 1 = maximum 1 0 = minimum 0 1 = updated 0 0 = not_updated
13:12	R	TX_updated_Status_main	0x0	A value in this register gives the current status transmitted to the Link Partner for the Main-Tap 1 1 = maximum 1 0 = minimum 0 1 = updated 0 0 = not_updated
11	R	Reserved	0x0	Reserved

10:8	R	LT_STATE_PRE	0x0	A value in this register gives the Coefficient Update State Machine for the Pre-Tap.
				IDLE = 3'b000;
				NOT_UPDATED = 3'b001;
				UPDATE_COEFF = 3'b011;
				MAXIMUM = 3'b010;
				UPDATED = 3'b110;
				MINIMUM = 3'b100;

7	R	Reserved	0x0	Reserved

6:4	R	LT_STATE_POST	0x0	A value in this register gives the Coefficient Update State Machine for the Post-Tap.
				IDLE = 3'b000;
				NOT_UPDATED = 3'b001;
				UPDATE_COEFF = 3'b011;
				MAXIMUM = 3'b010;
				UPDATED = 3'b110;
				MINIMUM = 3'b100;

3	R	Reserved	0x0	Reserved

2:0	R	LT_STATE_MAIN	0x0	A value in this register gives the Coefficient Update State Machine for the Main-Tap.
				IDLE = 3'b000;
				NOT_UPDATED = 3'b001;
				UPDATE_COEFF = 3'b011;
				MAXIMUM = 3'b010;
				UPDATED = 3'b110;
				MINIMUM = 3'b100;

7.1.1.2.23 Received Coefficient Status Register—(0x16) (Ask a Question)

The following table lists the received coefficient status register description.

Table 7-40. Received Coeff Status Register

Bit	Type	Function	Default	Description
31:16	NEW_RCVD_COEFF			A value in this register gives the new coefficient received from the Link Partner.
31:30	R	Reserved	0x0	Reserved Transmitted as 0, ignored on reception
29	R	Preset	0x0	1 = Pre-set coefficients 0 = Normal operation
28	R	Initialize	0x0	1 = Initialize coefficients 0 = Normal operation
27:22	R	Reserved	0x0	Reserved Transmitted as 0, ignored on reception
21:20	R	Post-Tap coefficient update	0x0	1 1 = Reserved 0 1 = Increment 1 0 = Decrement 0 0 = Hold
19:18	R	Main-Tap coefficient update	0x0	1 1 = Reserved 0 1 = Increment 1 0 = Decrement 0 0 = Hold
17:16	R	Pre-Tap coefficient update	0x0	1 1 = Reserved 0 1 = Increment 1 0 = Decrement 0 0 = Hold
15:0	NEW_RCVD_STATUS			A value in this register gives the new status received from the Link Partner.
15	R	Receiver Ready	0x0	1 = The remote receiver has determined that the training is completed and is prepared to receive data. 0 = The remote receiver is requesting that training continue.
14:6	R	Reserved		Reserved Transmitted as 0, ignored on reception
5:4	R	Post-Tap coefficient value	0x0	1 1 = Maximum 1 0 = Minimum 0 1 = updated 0 0 = Not Updated
3:2	R	Main-Tap coefficient value	0x0	1 1 = Maximum 1 0 = Minimum 0 1 = updated 0 0 = Not Updated
1:0	R	Pre-Tap coefficient value	0x0	1 1 = Maximum 1 0 = Minimum 0 1 = updated 0 0 = Not Updated

7.1.1.2.24 TX Coefficient Configuration Register—(0x18) (Ask a Question)

The following table lists the TX coefficient configuration register description.

Table 7-41. TX Coeff Configuration Register

Bit	Type	Function	Default	Description
31:12	R	Reserved	0x0	Reserved

...........continued

Bit	Type	Function	Default	Description
11	R/W	Reserved	0x0	Reserved
10	R/W	tx_hold_pre	0x0	If “1”, then hold coefficient is sent to the Link Partner for pre-cursor
9	R/W	tx_dec_pre	0x0	If “1”, then the decrement coefficient is sent to the Link Partner for pre-cursor
8	R/W	tx_inc_pre	0x0	If “1”, then the increment coefficient is sent to the Link Partner for pre-cursor
7	R/W	Reserved	0x0	Reserved
6	R/W	tx_hold_post	0x0	If “1”, then hold coefficient is sent to the Link Partner for post-cursor
5	R/W	tx_dec_post	0x0	If “1”, then the decrement coefficient is sent to the Link Partner for post-cursor
4	R/W	tx_inc_post	0x0	If “1”, then the increment coefficient is sent to the Link Partner for post-cursor
3	R/W	Reserved	0x0	Reserved
2	R/W	Tx_hold_main	0x0	If “1”, then hold coefficient is sent to the Link Partner for main-cursor.
1	R/W	tx_dec_main	0x0	If “1”, then the decrement coefficient is sent to the Link Partner for main-cursor.
0	R/W	tx_inc_main	0x0	If “1”, then the increment coefficient is sent to the Link Partner for main-cursor.

7.1.1.2.25 PRBS Error Word Count Register—(0X1F) (Ask a Question)

The following table lists the PRBS error word count register description.

Table 7-42. PRBS_ERROR_WORD_CNT Register

Bit	Type	Function	Default	Description
31:0	R	PRBS_ERROR_WORD_CNT	0x0	A value in this register gives the value of PRBS error count in terms of words.

7.1.1.2.26 10GBaseKR Status Register—(0x26) (Ask a Question)

The following table lists the 10GBaseKR status register description.

Table 7-43. 10GBaseKR Status Register

Bit	Type	Function	Default	Description
31:7	R	Reserved	0x0	Reserved
6	RW	RX calibration done	0x0	Write '1' followed by '0' to clear bit [2] (RX calibration request).
5	R	Request TX equalization	0x0	‘1’, indicates that the local device is responded for remote receiver RX calibration request. Set all bits to '1' of TX equlization register—(0xF) to clear this bit.
4	R	Signal detect	0x0	‘1’, indicates that the both local and remote receivers are ready. This bit gets cleared on reset.
3	R	Training fail	0x0	‘1’, indicates that 500 ms of time is expired during the Link Training. This bit gets cleared on reset.
2	R	Request RX calibration	0x0	'1' indicates that the remote receiver responded for calibration request sent by local device. This bit clears on writing '1' to Bit[6] of this register.
1	R	Reserved	0x0	Reserved
0	R	Auto-Negotiation good link check	0x0	‘1’, indicates that the Auto-Negotiation is completed. This bit gets cleared on reset.

7.1.1.3 Transmit Control Registers (Ask a Question)

Table 7-44. Transmit Control Registers

Address Offset PADDR [7:2]	Register [15:0]
0x0	Data Transmit Control Register
0x1	IP version register
0x2	FEC User configuration Register

7.1.1.3.1 Data Transmit Control Register—(0x00) (Ask a Question)

Table 7-45. Data Transmit Control Register

Bit	Type	Function	Default	Description
31:7	R/W	Reserved	0x00	—
6	R/Wc	RX_Reset	0	Soft reset bit for PCS reset 1—RX Logic is reset. 0—RX Logic is not reset. Self Clearing Bit
5	R/Wc	TX_Reset	0	Soft reset bit for PCS reset 1—TX Logic is reset. 0—TX Logic is not reset Self Clearing Bit
4	R/W	XCVR_LOS	0	Loss of sync signal to the XCVR 1—LOS signal is enabled, XCVR is lock to reference. 0—LOS signal is disabled, XCVR is lock to data.
3:2	R/W	Reserved	0x0	Reserved
[1:0]	R/W	PMA Data Select	0x0	PMA TX Data Select Used to select the TX raw data from the TX clause blocks to the SerDes Interface. 2'b00—PCS sublayer Clause 49 data is transmitted to the SerDes Interface. 2'b10—Auto-Negotiation block, PCS sublayer Clause 73 data is transmitted to the SerDes Interface. 2'b11—Link Training block, PCS sublayer Clause 72 data is transmitted to the SerDes Interface. 2'b01—Reserved

7.1.1.3.2 IP Version Register (Ask a Question)

The following table lists the data IP version register description.

Table 7-46. IP Version Register

Bit	Type	Function	Description
31:16	R	Major Version	This field provides the majorversion of the IP.
15:8	R	Minor Version	This field provides the minorversion of the IP.
7:0	R	Sub Version	This field provides the sub version of the IP.

7.1.1.3.3 FEC User Configuration Register (Ask a Question)

The following table lists the FEC user configuration register description.

Table 7-47. FEC User Configuration Register

Bit	Type	Function	Default	Description
31:1	R	Reserved	0x00	—
0	R	FEC_Configured	0	User has configured FEC Logic as part of the IP 0—FEC Not Configured 1—FEC is Configured This register is read by the software to enable the FEC during the AN.

7.1.1.4 Receive Status Registers (Ask a Question)

The following table lists the receive status register description.

Table 7-48. Receive Status Registers

Address Offset PADDR [7:2]	Register [15:0]
0x0	RX Status Register

7.1.1.4.1 RX Status Register—(0x00) (Ask a Question)

The following table lists the RX status register description.

Table 7-49. Receive Status Register

Bit	Type	Function	Default	Description
31:1	R/W	Reserved	0x00	—
0	R	PCS49 status	0	Receive Status Signal 1—Receiver has attained the Block Lock 0—Receiver has not attained the Block Lock

8. Embedded Software Support (Ask a Question)

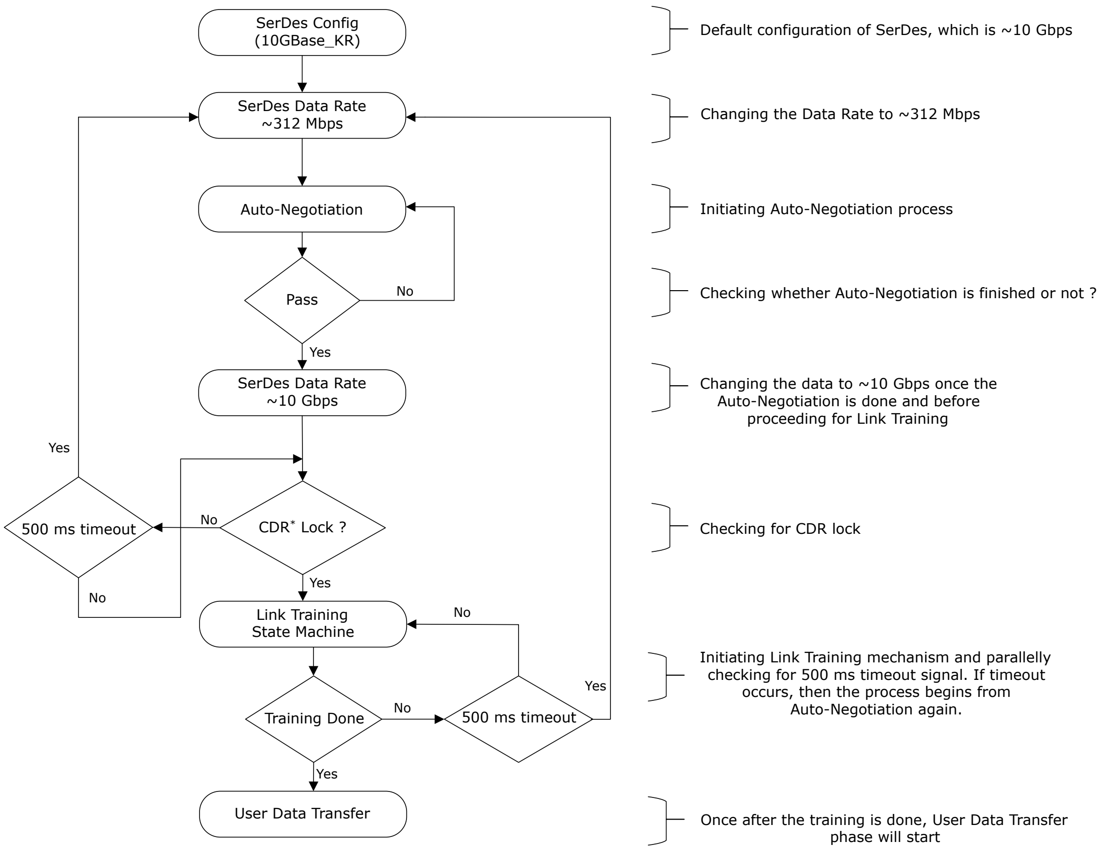
The Core10GBaseKR_PHY driver is responsible for configuring the transceiver registers, DRI registers, implementation of Auto-Negotiation, and Link Training algorithms. The GitHub repository provides access to the documentation for the driver and generate sample projects that illustrate how to use the driver.

The Mi-V Soft RISC-V embedded software driver is available on Mi-V-Soft-RISC-V/platform.

The Mi-V Soft RISC-V SoftConsole example project is available on Mi-V-Soft-RISC-V/miv-rv32-bare-metal-examples.

Following figure shows the sequence that needs to be followed for SerDes configuration, Auto-Negotiation, and Link Training.

Figure 8-1. 10GBaseKR Sequence


SerDes Config (10GBase_KR)
SerDes Data Rate ~312 Mbps
Auto-Negotiation
Pass
No
Yes
SerDes Data Rate ~10 Gbps
Yes
500 ms timeout
No
CDR* Lock ?
No
Yes
Link Training State Machine
No
Training Done
No
500 ms timeout
Yes
Yes
User Data Transfer
Default configuration of SerDes, which is ~10 Gbps
Changing the Data Rate to ~312 Mbps
Initiating Auto-Negotiation process
Checking whether Auto-Negotiation is finished or not ?
Changing the data to ~10 Gbps once the Auto-Negotiation is done and before proceeding for Link Training
Checking for CDR lock
Initiating Link Training mechanism and parallelly checking for 500 ms timeout signal. If timeout occurs, then the process begins from Auto-Negotiation again.
Once after the training is done, User Data Transfer phase will start

*CDR - Clock Data Recovery

Important: The SerDes registers configured as part of link training through the embedded software might not be the ideal values across all the channel losses. These values need to be changed by the user as per the channel loss of the system setup. For more information about the transceiver registers, see the PolarFire FPGA and PolarFire SoC FPGA Transceiver User Guide.

Auto-Negotiation algorithm is implemented in hardware in accordance with the *Clause 73 of IEEE 802.3 standard*. For more information about Arbitration state diagram, see Figure 73-11. Embedded software is responsible for initiating Auto-Negotiation. The firmware advertises the FEC feature (ability and request) when FEC block is enabled through the IP configurator. For more details about the FEC feature, see *section 73.6.5 of IEEE 802.3 standard* . Link Training algorithm is implemented in hardware in accordance with the *IEEE specification Clause-72*. For more information about Training state diagram, see Figure 72-5. Embedded software is responsible for initiating Link Training post Auto-Negotiation is complete.

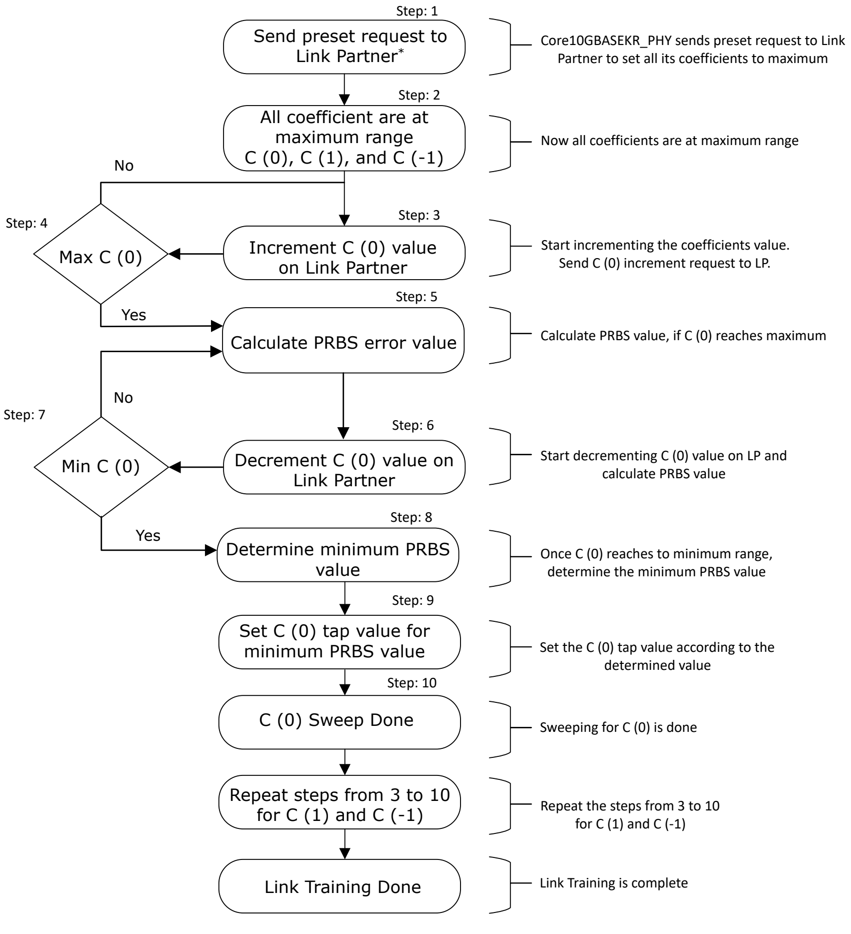
The local receiver training (TRAIN_LOCAL) specified in the *Figure 72-5* as specified in *Clause 72 of IEEE 802.3 standard* is implemented in the embedded software.

8.1 Link Training Procedure (Ask a Question)

This section describes the Link training procedure.

The following figure shows the sequence that is followed to train the local receiver. For more information, see 8.1.2. RX Calibration Algorithm.

Figure 8-2. RX Calibration Algorithm


Step: 1
Send preset request to Link Partner*
Core10GBASEKR_PHY sends preset request to Link Partner to set all its coefficients to maximum
Step: 2
All coefficient are at maximum range C (0), C (1), and C (-1)
Now all coefficients are at maximum range
No
Step: 3
Step: 4
Max C (0)
Increment C (0) value on Link Partner
Start incrementing the coefficients value. Send C (0) increment request to LP.
Yes
Step: 5
Calculate PRBS error value
Calculate PRBS value, if C (0) reaches maximum
No
Step: 7
Step: 6
Min C (0)
Decrement C (0) value on Link Partner
Start decrementing C (0) value on LP and calculate PRBS value
Yes
Step: 8
Determine minimum PRBS value
Once C (0) reaches to minimum range, determine the minimum PRBS value
Step: 9
Set C (0) tap value for minimum PRBS value
Set the C (0) tap value according to the determined value
Step: 10
C (0) Sweep Done
Sweeping for C (0) is done
Repeat steps from 3 to 10 for C (1) and C (-1)
Repeat the steps from 3 to 10 for C (1) and C (-1)
Link Training Done
Link Training is complete

*LP – Link Partner

8.1.1 Interpreting the 10GBaseKR Status Register (Ask a Question)

To perform link training, embedded software must handle the following requests set by the Core10GBaseKR_PHY IP in the interrupt status register (offset: 0x26). The embedded software should read this register and handle the following requests:

- RX Calibration Request
- TX Equalization Request
- Signal Detect
- Training Failure

8.1.2 RX Calibration Algorithm (Ask a Question)

The RX calibration algorithm is used to calibrate the transmitter tap settings of the link partner which allows the local receiver to receive ethernet packets with minimal error. When the REQUEST_RX_CALIBRATION bit is set, the embedded software must clear this status by setting the RX_CALIBRATION_DONE bit. Based on the status report field read from the Received Coefficient Status register (offset: 0x16), the TX Coefficient configuration register (offset: 0x18) must be configured with the next request to be sent to the link partner that is, increment or decrement. Setting the TX Coefficient configuration register updates the coefficient update field of the training frame sent to the Link Partner.

8.1.3 TX Equalization Algorithm and Transceiver Register Update (Ask a Question)

When the TX_EQUALIZATION bit is set, the embedded software should read the registers; TX New Main-TapRegister (offset: 0x11), TX New Post-Tap register (offset: 0x12) and TX New Pre-Tap register (offset: 0x13) and update the transceiver transmitter coefficient tap values. For more information on XCVR registers, see the *Dynamic Reconfiguration Interface (DRI) Handbook*. This Handbook can be downloaded from the Libero catalog. After configuring the transceiver transmitter tap coefficients, the embedded software should set value of the TX Equalization register (offset: 0xF) to 0xF to clear the interrupt status register's TX_EQUALIZATION bit.

8.1.4 Signal Detect (Link Training Completion) (Ask a Question)

Local Receiver Lock Register (offset: 0x10) indicates the Local Receiver Ready. Link Partner Ready is indicated by the status report field. If both the Local Receiver Ready and the Link Partner Ready bits are set, then the Core10GBaseKR_PHY IP sets the SIGNAL_DETECT bit indicating to the embedded software that link training is completed.

8.1.5 Training Failure (Ask a Question)

The training fail bit at register (offset: 0x26) indicates that the training has timed out, then the embedded software should reinitiates the auto-negotiation and Link Training.

Important: The link training timer timeout is specified in the Max Wait Timer Configuration register (offset: 0x1).

9. Additional References (Ask a Question)

This section provides a list for additional information.

For updates and additional information about the software, devices, and hardware, visit the **Intellectual Property** pages on the Microchip FPGAs and PLDs website.

9.1 Known Issues and Workarounds (Ask a Question)

The following are the known issues for the FEC feature:

- Validated for point-to-point Link
- Interop validation is not performed

10. Revision History (Ask a Question)

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

Table 10-1. Revision History

Revision	Date	Description
C	08/2023	The following is the list of changes in the revision C of the document: • Updated the document for Core10GBaseKR_PHY v3.0 • Added Table 1 and Table 2 • Updated the Table 2-3 for FEC signals • Updated timing constraints file in 6.1. Timing Constraint • Added a new section 7.1.1.3.3. FEC User Configuration Register
B	12/2022	The following is the list of changes in the revision B of the document: • Updated the Default value for Table 7-19, Table 7-20, Table 7-21, Table 7-22, Table 7-24, Table 7-25, Table 7-26, Table 7-27, Table 7-28, and Table 7-29 • Added IP version register row in Table 7-44 • Added IP version register description in 7.1.1.3.1. Data Transmit Control Register—(0x00) • Renamed the Receive Control Register section to Receive Status Register • Added new section 8.1. Link Training Procedure • Updated timing constraints in 6.1. Timing Constraint • Updated 1. Device Utilization and Performance
A	09/2022	Initial Release

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic

ISBN: 978-1-6683-2927-6

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-72884388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820