

40G/50G High Speed Ethernet Subsystem v3.3

产品指南

Vivado Design Suite

PG211 (v3.3) 2022 年 11 月 3 日

本文档为英语文档的翻译版本，若译文与英语原文存在歧义、差异、不一致或冲突，概以英语文档为准。译文可能并未反映最新英语版本的内容，故仅供参考，请参阅最新版本的英语文档获取最新信息。

赛灵思矢志不渝地为员工、客户与合作伙伴打造有归属感的包容性环境。为此，我们正从产品和相关宣传资料中删除非包容性语言。我们已发起内部倡议，以删除任何排斥性语言或者可能固化历史偏见的语言，包括我们的软件和 IP 中嵌入的术语。虽然在此期间，您仍可能在我们的旧产品中发现非包容性语言，但请确信，我们正致力于践行革新使命以期与不断演变的行业标准保持一致。如需了解更多信息，请参阅此[链接](#)。



目录

第 1 章：简介.....	4
功能特性.....	4
IP 相关信息.....	5
第 2 章：概述.....	6
按设计进程浏览内容.....	6
子系统概述.....	6
功能特性总结.....	7
应用.....	8
许可和订购.....	8
第 3 章：产品规格.....	10
典型操作方法.....	11
统计数据收集.....	12
可测试性功能.....	12
标准.....	12
性能和资源使用情况.....	12
时延.....	12
端口描述.....	13
自动协商 (AN) 和链路训练 (LT).....	36
链路训练描述.....	44
使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试.....	46
前向纠错 (FEC).....	47
PCS 变体.....	47
AXI4-Lite 寄存器空间.....	55
第 4 章：利用子系统进行设计.....	112
通用设计指南.....	112
时钟设置.....	113
LogiCORE 设计示例时钟设置和复位.....	117
适用于 GTM 的 SerDes 数据映射.....	126
适用于 40G/50G 子系统的 IEEE PTP 1588v2.....	129
IEEE PTP 1588v2 功能描述.....	134
RS-FEC 支持.....	138
状态/控制接口.....	141
第 5 章：设计流程步骤.....	144
自定义和生成子系统.....	144

约束子系统.....	152
仿真.....	153
综合与实现.....	156
第 6 章：设计示例.....	157
设计示例层级.....	157
设计示例层级（设计示例中包含 GT）.....	159
用户接口.....	162
核 xci 顶层端口列表.....	163
双工操作模式.....	205
运行时可切换.....	206
共享逻辑实现.....	209
AXI4-Lite 接口实现.....	210
AXI4-Lite 接口用户逻辑.....	211
IEEE 第 91 条 (RS-FEC) 集成.....	214
第 7 章：测试激励文件.....	216
附录 A：升级.....	217
从 v2.2 到 v2.3 的更改.....	217
从 v2.1 到 v2.2 的更改.....	218
从 v2.0 到 v2.1 的更改.....	218
从 v2.0（2016 年 10 月 5 日）到 v2.0（2016 年 11 月 30 日）的更改.....	219
从 v1.1 到 v2.0 的更改.....	220
从 v1.0 到 v1.1 的更改.....	221
附录 B：调试.....	224
在 Xilinx.com 上寻求帮助.....	224
调试工具.....	225
仿真调试.....	225
硬件调试.....	227
接口调试.....	229
协议调试.....	230
附录 C：暂停处理接口.....	233
TX 暂停生成.....	233
RX 暂停终止.....	234
附录 D：附加资源与法律声明.....	238
赛灵思资源.....	238
Documentation Navigator 与设计中心.....	238
参考资料.....	238
修订历史.....	239
请阅读：重要法律声明.....	242

简介

赛灵思 High Speed Ethernet IP 子系统可通过物理编解码子层 (PCS) 或独立 PCS 来实现 40G 或 50G 以太网媒体访问控制器 (MAC)。

功能特性

- 根据 25G 以太网联盟（以太网技术联盟）的 Schedule 3 所定义的以太网 50 Gb/s 运行要求而设计。请参阅页面底部的注释。
- 根据 IEEE 802.3 第 82 条《IEEE 以太网标准》（[IEEE 802.3-2015 标准](#)）定义的 40 Gb/s 运行要求而设计。
- 包含完整的以太网 MAC 和 PCS 功能或独立 PCS。
- 数据包导向的简单用户接口。
- 低时延模式。
- 全方位的统计数据收集。
- 对应所有主要功能指示符的状态信号。
- 随顶层封装一起交付，包括功能性收发器封装、IP 网表、测试脚本样本以及 Vivado® 设计工具编译脚本。
- 可选付费型 AN/LT/KR 前向纠错 (FEC) 功能。
- 可选 KP4 FEC 功能（适用于 GTM 器件）。

注释：要访问 25G 规范，请访问[以太网技术联盟](#)网站。

IP 相关信息

LogiCORE™ IP 相关信息表	
子系统规格	
支持的器件系列	Versal® ACAP Zynq® UltraScale+™ RFSoc Zynq® UltraScale+™ MPSoC Virtex® UltraScale+™ Kintex® UltraScale+™ Kintex® UltraScale™ Artix® UltraScale+™
支持的用户接口	适用于 50 Gb/s 的 128 位跨接包 AXI4-Stream 适用于 40 Gb/s 的 128 位跨接包或 256 位 AXI4-Stream
资源	性能和资源使用情况网页
随子系统提供	
设计文件	加密 RTL
设计示例	Verilog
测试激励文件	Verilog
约束文件	赛灵思设计约束 (XDC)
仿真模型	Verilog
支持的软件驱动程序	不适用
经过测试的设计流程 ¹	
设计输入	Vivado® Design Suite
仿真	如需了解受支持的仿真器的相关信息，请参阅 赛灵思设计工具：版本说明指南 。
综合	Synopsys 或 Vivado 综合
支持	
版本说明和已知问题	主答复记录： 54690
所有 Vivado IP 更改日志	Vivado IP 主更改日志： 72775
赛灵思技术支持网页	

注释：

1. 如需了解受支持的第三方工具版本的相关信息，请参阅[赛灵思设计工具：版本说明指南](#)。
2. 请联系赛灵思技术支持以了解设计要求。

概述

按设计进程浏览内容

赛灵思文档按一组标准设计进程进行组织，以便帮助您查找当前开发任务相关的内容。所有 Versal® ACAP 设计进程的对应[设计中心](#)和[设计流程助手](#)资料均可在 [Xilinx.com](#) 网站上找到。本文档涵盖了以下设计进程：

- 硬件、IP 和平台开发：为硬件平台创建 PL IP 块、创建 PL 内核、功能仿真以及评估 Vivado® 时序收敛、资源使用情况和功耗收敛。还涉及为系统集成开发硬件平台。本文档中适用于此设计进程的主题包括：
 - [端口描述](#)
 - [AXI4-Lite 寄存器空间](#)
 - [配置寄存器](#)
 - [状态寄存器](#)
 - [统计数据计数器](#)
 - [时钟设置](#)
 - [自定义和生成子系统](#)
 - [第 6 章：设计示例](#)
 - [附录 B：调试](#)

子系统概述

赛灵思 40G/50G High Speed Ethernet 子系统可实现 40G/50G 以太网媒体访问控制器 (MAC) 模块，其中包含 40G/50G PCS 或独立 40G/50G PCS。

40G/50G High Speed Ethernet 子系统是遵循适用于 50 Gb/s 运行速率的 25G 和 50G 以太网联盟规范 r1.6 版本的 Schedule 3 以及适用于 40 Gb/s 的 IEEE 802.3 来设计的；它已经过硬件验证，可支持系统设计师快速且无风险地搭建系统，来实现 40G/50G 以太网协议。

本指南还对 40G/50G High Speed Ethernet 子系统进行了详细描述，并提供了将 40G/50G High Speed Ethernet 子系统集成到用户设计中所需的信息。本文档假定您熟悉 IEEE 802.3-2015 标准协议以及 FPGA 设计和方法论。如需了解详细信息，请参阅 [第 3 章：产品规格](#)。如需了解特定于赛灵思器件平台的信息，请参阅[赛灵思技术支持](#)。另请参阅《IEEE 以太网标准》（[IEEE 802.3-2015 标准](#)）。

功能特性总结

- 支持自定义前导码
- 可编程包间间隔 (IPG)
- 用于 40 Gb/s 和 50 Gb/s 运行的简单包导向 128 位跨接包 AXI4-Stream 接口
- 用于 40 Gb/s 运行的可选 256 位常规 AXI4-Stream 接口
- 可选第 74 条前向纠错 (FEC)
- 50G FEC (以太网联盟 Schedule 3 规范, 基于 IEEE 802.3 第 91 条)
- 可选 1588v2 PTP 单步和双步时间戳
- 可选自动协商和链路训练

表 1: 特性兼容性矩阵

变体	MAC	PCS	128 位跨接包 ²	256 位	暂停处理	自动协商和链路训练 ⁴	第 74 条 FEC	第 91 条 RS-FEC	软核 FEC RS ⁴	IEEE 1588 时间戳
			AXI4-Stream							
低时延 40G MAC + PCS	X	X	X	X						
40G MAC + PCS	X	X	X	X	X	X	X			X
40G 仅限 PCS		X				X	X			X ¹
低时延 50G MAC + PCS	X	X	X							
50G MAC + PCS	X	X	X		X	X	X	X	X	X
50G 仅限 PCS		X				X	X	X	X	X ¹
运行时可 切换 40/50G MAC+PCS ³	X	X	X	-	-	X	X	X		X
运行时可 切换 40/50G 仅 限 PCS ³	-	X	-	-	-	X	X	X		X ¹

注释:

1. 双步时间戳只受仅限 PCS 配置支持。
2. UltraScale™/ UltraScale+™ 速度等级 -1 不支持 40G 128 位 AXI4-Stream 接口。建议对速度等级 -1 使用 256 位 AXI4-Stream 接口。
3. GTM 型收发器不支持自动协商和链路训练, 未来版本中将予以支持。
4. Versal 器件不支持自动协商、链路训练、控制端口和状态端口。

应用

赛灵思 40G/50G High Speed Ethernet 子系统设计旨在作为网络接口，以供要求极高比特率的应用使用，例如：

- 以太网交换机
- IP 布线器
- 数据中心交换机
- 通信设备

互连器件能否以 50 Gb/s 以太网速率运行对于下一代数据中心网络尤其重要，这主要体现在：

- 为了跟上不断增加的 CPU 和存储带宽需求，机架式或刀片式服务器必须支持来自其网络接口卡 (NIC) 或板载 LAN (LOM) 网络端口的聚合吞吐量超过 10 Gb/s (单通道) 或 20 Gb/s (双通道)。
- 由于端点带宽增加，架顶式 (TOR) 或刀片式交换机的上行链路需要从 40 Gb/s (四通道) 升级到 100 Gb/s (四通通道)，同时最好保留同样的每通道分接能力。
- 由于预计将采用 100GBASE-CR4/KR4/SR4/LR4，因此 SerDes 和布线技术均已在开发和部署中，以支持每个物理通道、双轴线缆或光纤 25 Gb/s 的传输速率。

许可和订购

根据[赛灵思核许可协议](#)，提供 40G/50G Ethernet IP 核。该模块作为 Vivado® Design Suite 的一部分提供。必须购买一个或多个核许可证才能获取仿真和硬件中的所有核功能的完整访问权限。

注释： 40G/50G Ethernet MAC + BASE-R、XLAUI/LAUI 和 40GBASE-KR4/50GBASE-KR2 IP 选项需要单独的器件编号。

如需了解有关 40G/50G Ethernet 核定价和可用性的更多信息，请联系您当地的[赛灵思销售代表](#)。如需了解更多信息，请访问 [40G/50G Ethernet Subsystem 页面](#)。

如需了解其它赛灵思 LogiCORE™ 模块，请访问[赛灵思知识产权页面](#)。

要购买其中任何 IP 核，请联系您当地的[赛灵思销售代表](#)，并提供下表中相应的器件编号。

表 2：订购信息

描述	器件编号	许可证密钥
40G/50G Ethernet MAC + BASE-R ¹	EF-DI-50GEMAC-PROJ EF-DI-50GEMAC-SITE	l_eth_mac_pcs
50GBASE-KP (RS-FEC (544 和 514) AN/LT) ² 50GBASE-KR2 (RS-FEC (528 和 514) , AN/LT) 40GBASE-KR4 (第 74 条 FEC, AN/LT) 50G RS-FEC (独立或 EF-DI-50GEMAC 附加项) XLAUI/ LAUI (40GBASE-R/50GBASE-R) 注释： 除了 EF-DI-50GBASE-KR2-xxxx 器件编号外，如果还需要 MAC，请订购 EF-DI-50GEMAC-xxxx。	EF-DI-50GBASE-KR2-PROJ EF-DI-50GBASE-KR2-SITE	l_eth_basekr ieee802d3_50g_rs_fec l_eth_baser

表 2：订购信息 (续)

描述	器件编号	许可证密钥
XLAUI/LAUI PCS/PMA (40GBASE-R/50GBASE-R) 注释： AN/LT/FEC/MAC 不包含在此器件编号中。如果需要这些，请参阅以上内容。	EF-DI-LAUI-PROJ EF-DI-LAUI-SITE	l_eth_baser

注释：

1. 要访问独立 PCS/PMA (40GBASE-R/50GBASE-R)，必须订购 EF-DI-LAUI-XXXX 或 EF-DI-50GBASE-KR2-xxxx。
2. 用于 50GBASE-CR2、50GBASE-KR2、50GBASE-SR2 和 50GBASE-FR2 应用。50G RS-FEC (544 和 514) 可用作 GTM 收发器的 UltraScale+ 58G 器件中的硬化 IP，无需额外费用。

许可证检查器

如果 IP 需要许可证密钥，则必须验证密钥。Vivado® 设计工具设置有多个许可证检查点，旨在通过此流程对获得许可的 IP 进行门控。如果许可证检查成功，则可以继续生成 IP。否则，系统会因错误而停止生成。许可证检查点由以下工具强制执行：

- Vivado 综合
- Vivado 实现
- write_bitstream (Tcl 命令)



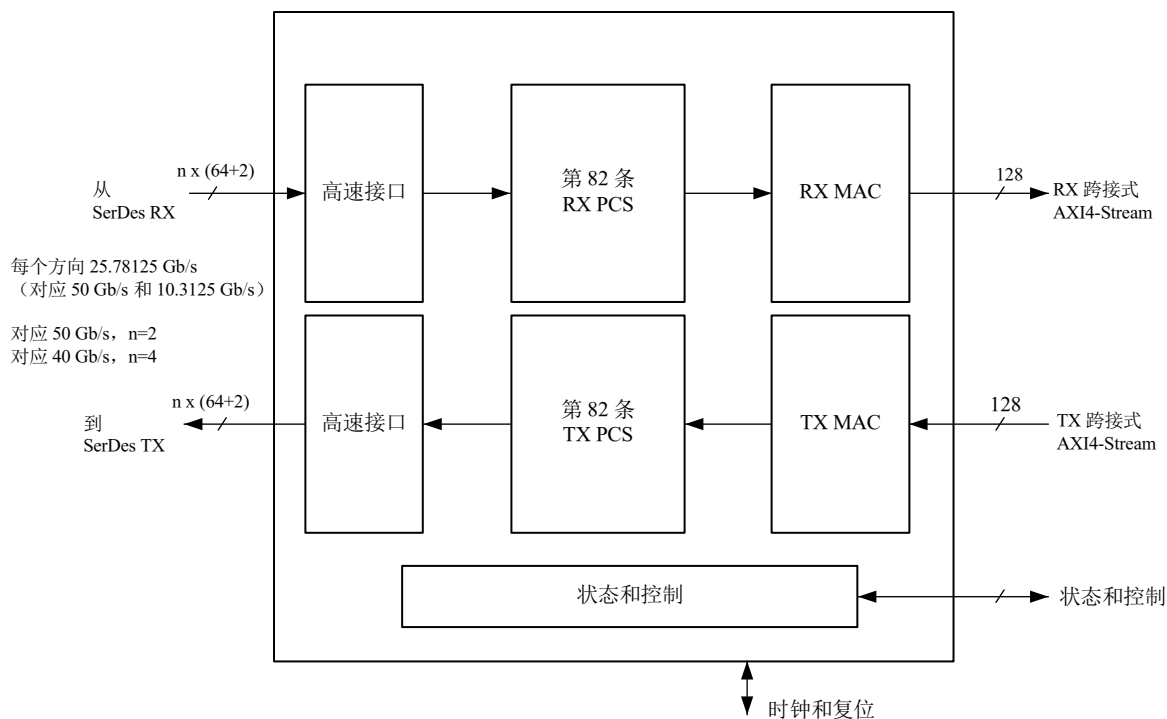
重要提示！ 在检查点处忽略 IP 许可证级别。测试会确认是否存在有效的许可证。它不检查 IP 许可证级别。

产品规格

40G/50G High Speed Ethernet IP 子系统支持与 High Speed Ethernet 核建立易于使用的连接。根据配置，该子系统可创建接口端口、例化 40G/50G High Speed Ethernet Subsystem、例化高速串行和解串器 (SerDes) 块、提供相应的时钟信号并连接至 AXI4-Stream 用户侧接口。

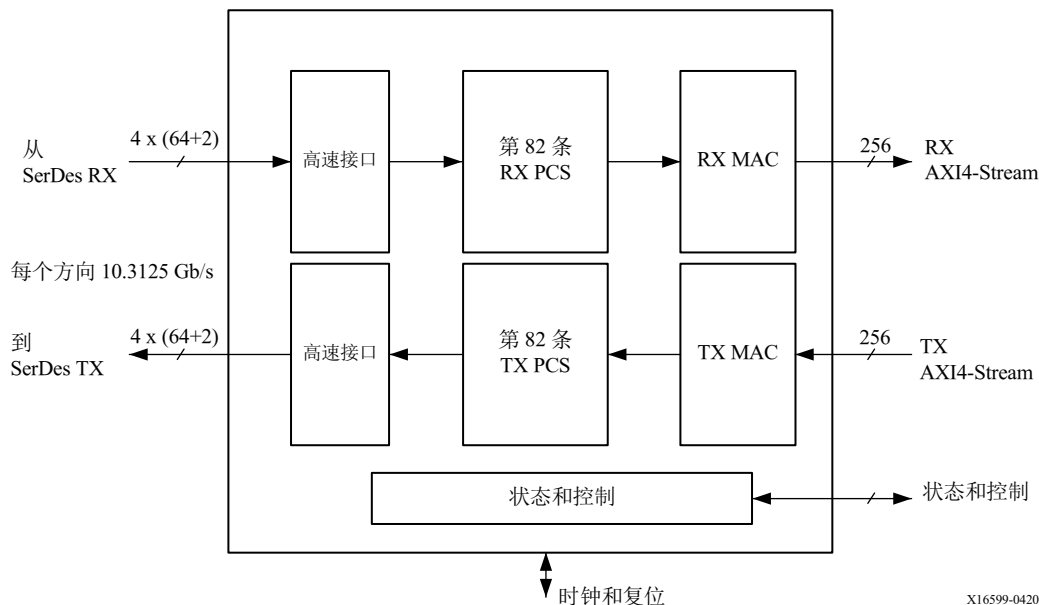
40G/50G High Speed Ethernet 子系统的模块框图如下图所示。右侧为用户接口，左侧为外部器件接口。

图 1: 含 128 位跨接式 AXI4-Stream 的 40G/50G High Speed Ethernet Subsystem



X23911-042021

图 2：含 256 位 AXI4-Stream 的 40G High Speed Ethernet Subsystem



PCS 架构取决于发射器件在通过多个（相对）低速的物理接口来发射数据包时所采用的数据包各部分的分发（或分割）方式。随后，接收器件 PCS 层负责对各不同部分进行分割，并重构数据包，然后再将其交付给 Ethernet MAC 块。接收器 PCS 层也必须对来自不同物理接口的数据进行去歪斜，因为这些接口在整个网络内进行传输时可能产生不同延迟。此外，该核还负责处理接收到的所有 PCS 通道间的 PCS 通道交换，以使 40G/50G High Speed Ethernet 子系统可供所有光学传输系统使用。

该核的 PCS 和 Ethernet MAC 层按接口的最大线速率来运行，并且已经过最优化以在赛灵思 FPGA 内运行。PCS 层包含扰码/解扰和 64B/66B 编码器/解码器，并以 40G/50G 线速率全速运行。Ethernet MAC 块包含经最优化的高速帧校验序列 (FCS) 生成和校验模块。除检查数据包的 FCS 完整性外，FCS 模块还支持以 40G/50G 线速率全速插入和删除数据包的 FCS 字节（可选）。

控制和状态 (Control and Status) 块可提供多个统计数据计数器用于监控数据流量。此外，40G/50G High Speed Ethernet Subsystem 的状态接口提供了有关总体接口、每个物理接口以及每条 PCS 通道的运行状况详细信息。此状态信息包含同步报头对齐、PCS 对齐和 PCS 去歪斜状态信息。

典型操作方法

核处理所有协议相关功能，以便与另一个器件的 PCS 和以太网 MAC 接口进行通信。这包括握手、同步和纠错。您通过 AXI4-Stream TX 接口提供包数据，并通过 AXI4-Stream RX 接口接收包数据。如需了解详细描述，请参阅 [AXI4-Stream 接口](#)。

40G/50G High Speed Ethernet 子系统设计尽可能提升灵活性，适用于许多不同应用。RX 路径仅用于执行必要的流水打拍以满足所需操作的要求，而不执行任何缓冲。接收数据以直通方式直接传递到用户接口，以便您灵活实现任何所需的缓冲方案。并且，核 TX 路径包含单一流水线，其中仅含提供可靠的直通传递操作所需的最少量的缓冲。

统计数据收集

40G/50G High Speed Ethernet 子系统可提供灵活且用户友好的统计数据收集机制。对于每项受支持的统计数据，该核提供 1 个输出信号（或总线），用于指示给定时钟周期内统计数据的增量值。此机制支持您选择系统中所需的统计数据，同时避免了一整套计数器所造成的成本开销。此外，更重要的是，您可实现系统所需的任何计数器和统计数据收集机制。例如，您可以根据需要构建 32 位或 64 位，或者根据需要实现读取时清零计数器或饱和计数器。并且还提供了可选的 AXI4-Lite 寄存器实现，其中包含统计数据计数器。如需了解可选 AXI4-Lite 实现的详细描述，请参阅 [第 6 章：设计示例](#)。

对于 TX 统计数据，有效包数定义为不含 FCS 或任何其它错误的包的数量，无效包数定义为含 FCS 或任何其它错误的包的数量。

对于 RX 统计数据，有效包数定义为不含 FCS 或任何其它错误（包括长度错误）的包的数量，无效包数定义为含 FCS 或任何其它错误的包的数量。长度字段错误包括包长度字段错误、包大小过大或包大小过小。

注释：对于 Versal® 器件，始终启用 AXI4-Lite 接口并提供额外 GUI 选项以包含统计数据计数器。

可测试性功能

该核可根据第 82.2.10 条（测试模式生成器）和第 82.2.17 条（测试模式校验器）的定义，实现测试模式生成和校验。如需了解更多详情，请参阅《IEEE 以太网标准》（[IEEE 802.3-2015 标准](#)）。

标准

40G/50G High Speed Ethernet 子系统是根据 25G 和 50G 以太网联盟规范 r1.6 版的 Schedule 3（适用于 50 Gb/s 运行速率）和 IEEE 802.3（适用于 40 Gb/s 运行速率）而设计的。

性能和资源使用情况

如需获取有关性能与资源使用情况的完整详情，请访问[性能与资源使用情况网页](#)。

时延

下表提供了针对 40G/50G IP 核的低时延设计测量所得的时延信息。此信息由该核的 RX 和 TX 时延组合而成，不包括收发器中的时延。

表 3: 40G/50G IP 核的低时延设计的时延结果

核	总时延 (ns)	TX 时延 (ns)	RX 时延 (ns)	用户总线宽度 (位)	SerDes 数据宽度 (位)	核时钟频率 (MHz)
40G MAC_PCS	99.2	35.2	64	128	32	312.5
50G MAC+PCS	84.5	25.6	58.8	128	64	390.625
40G PCS	196.5	97.5	99	128	32	312.5
50G PCS	161.8	79.2	82	128	64	390.625
40G MAC+PCS	153	64	89	256	32	312.5

注释：

1. 这些数值包含 RX 和 TX 互连结构逻辑，但不包含 GT。
2. MAC+PCS 数值适用于不含 FIFO 的低时延。PCS/PMA 始终包含 FIFO，因此时延数值较高。

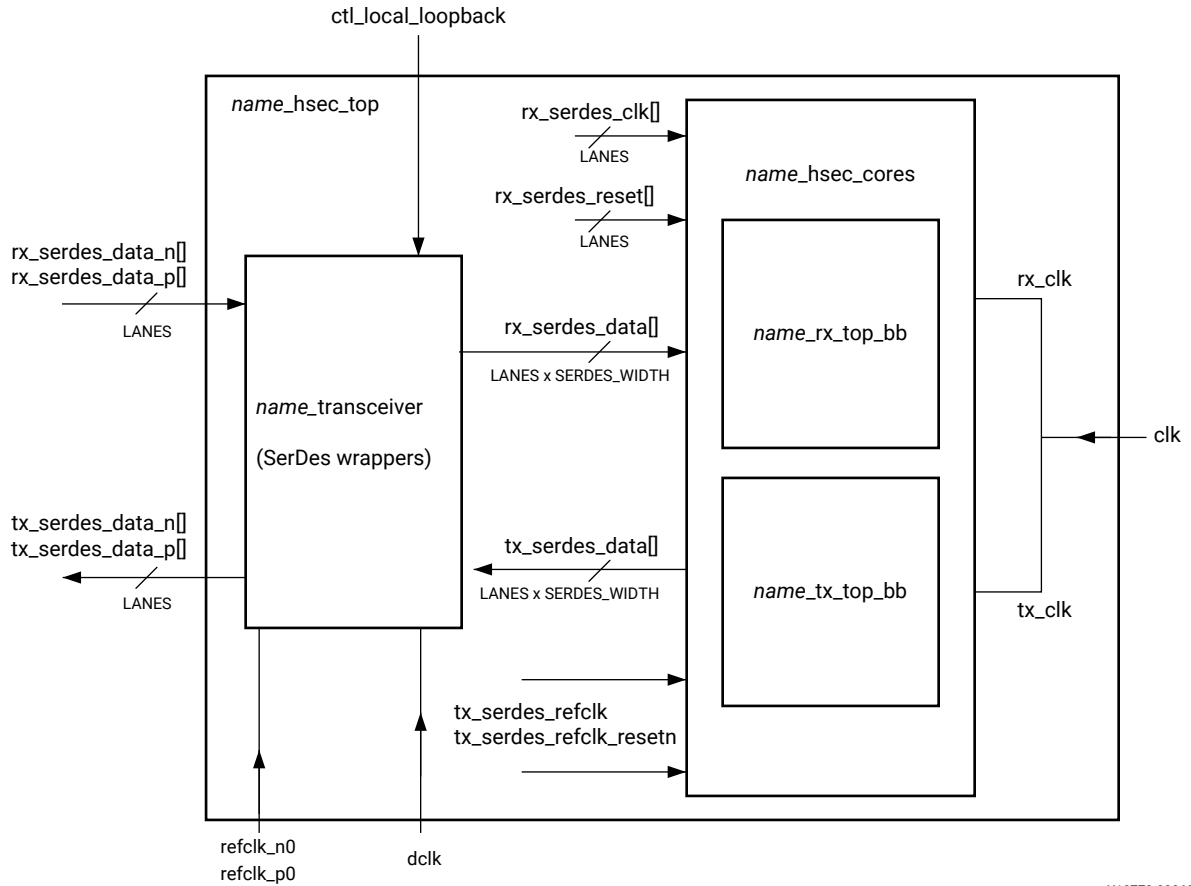
端口描述

下表列出了适用于 40G/50G 子系统的所有端口，包括用于 name_hsec_cores 层级上可选特性的端口。在 xci 顶层层级，端口进行了一些更改以包含例化的收发器核和其它共享逻辑。如需了解有关 XCI 层级端口列表的描述，请参阅 [第 6 章：设计示例](#) 的“核 xci 顶层端口列表”。如需了解有关其它端口的描述，请参阅“PCS 变体”以及“自动协商 (AN) 和链路训练 (LT)”。

下图显示了层级块之间的关系，并且展示了不同块在信号方面的差异。

注释： 在生成可选的 AXI4-Lite 寄存器时，其中一些端口可通过相应的寄存器（而非宽边总线）来访问。

图 3：端口列表层级



X18772-080621

以下小节描述了这些端口。VL_LANES 参数为 4，LANES 参数则是 2 (50 Gb/s) 和 4 (40 Gb/s)。对于 GTM，请参阅 [适用于 GTM 的 SerDes 数据映射](#)。

收发器端口

下表描述了收发器 I/O 端口。如需了解有关时钟域的信息，请参阅“时钟设置”主题。

表 4：收发器 I/O 端口列表

名称	I/O	描述
rx_serdes_data[LANES-1:0][64-1:0]	输入	来自 SerDes 宏的数据总线。存在 LANES rx_serdes_data 总线；每个 SerDes 通道对应一个总线，每个总线包含 64 位。按照定义，位 [64-1] 是 40G/50G High Speed Ethernet Subsystem 接收到的第一个位。位 [0] 是接收到的最后一个位。典型宽度是 64。
tx_serdes_data[LANES-1:0][64-1:0]	输出	通向 SerDes 宏的数据总线。存在 LANES tx_serdes_data 数据总线；每个 SerDes 通道对应一个总线，每个总线包含 64 位。按照定义，位 [64-1] 是 40G/50G High Speed Ethernet Subsystem 发射的第一个位。位 [0] 是发射的最后一个位。典型宽度是 64。
rx_serdes_clk[LANES-1:0]	输入	每个 SerDes 通道的恢复时钟。每个通道的 rx_serdes_data 总线均同步到该总线的对应位的上升沿。

表 4：收发器 I/O 端口列表 (续)

名称	I/O	描述
rx_serdes_reset[LANES-1:0]	输入	每个 RX SerDes 通道的复位。每个 SerDes 通道的恢复时钟都有 1 个与之关联的高电平有效复位。只要关联的恢复时钟未按正确频率运行，此信号就应断言有效。通常，此信号连接到锁相环 (PLL) 锁定信号。这是同步复位。
tx_serdes_refclk	输入	TX 数据路径的参考时钟。此时钟必须锁频到 tx_serdes_clk 输入。通常，用于驱动 TX SerDes 的参考时钟连接至此输入。
tx_serdes_refclk_reset	输入	TX 参考时钟的复位。只要 tx_serdes_refclk 输入未按正确频率运行，此信号就应断言有效。这是同步复位。

相关信息

时钟设置

AXI4-Stream 接口

本章节描述了 40G/50G 子系统的 AXI4-Stream 数据接口的连接方式。AXI4-Stream 接口遵循标准 Arm® AMBA 4 AXI4-Stream 协议 v1.0 规范，但广泛使用“user”信号。请参阅《AMBA AXI4-Stream 协议规范》(ARM IHI 0051A)，以获取该接口的详细描述。

40G/50G High Speed Ethernet 子系统可提供 128 位跨接包 AXI4-Stream 接口以供按 40 Gb/s 和 50 Gb/s 来运行。此外，对于只能以 40 Gb/s 运行的 IP，还可选择 256 位常规 AXI4-Stream 接口。

AXI4-Stream 接口 - 128 位跨接包接口

在 AXI4-Stream 接口中，以下各表描述了时钟/复位信号、接收接口信号和发射接口信号。

表 5：AXI4-Stream 接口 - 时钟/复位信号

名称	I/O	时钟域	描述
dclk	输入		此时钟必须为稳定且便于使用的时钟，例如，75 MHz。请参阅最新收发器指南以获取最新信息。
rx_reset	输入		RX 电路的复位。此信号为高电平有效（1 = 复位），并且必须保持高电平，直至 clk 达到稳定状态为止。40G/50G High Speed Ethernet Subsystem 负责处理将 rx_reset 输入同步到核中相应的时钟域的操作。这是同步复位。
refclk_n0	输入		对应收发器的差分参考时钟 (N)。
refclk_p0	输入		对应收发器的差分参考时钟 (P)。
tx_reset	输入		TX 电路的复位。此信号为高电平有效（1 = 复位），并且必须保持高电平，直至 clk 达到稳定状态为止。40G/50G High Speed Ethernet Subsystem 负责处理将 tx_reset 输入同步到核中相应的时钟域的操作。这是同步复位。

AXI4-Stream 接收接口

AXI4-Stream 接收接口类似于发射端，RX 数据对应于接收到的以太网帧。RX AXI 总线上其它信号的含义类似于 TX 总线上的信号。

下表显示了 AXI4-Stream 接收接口信号。

表 6: AXI4-Stream 接收接口信号

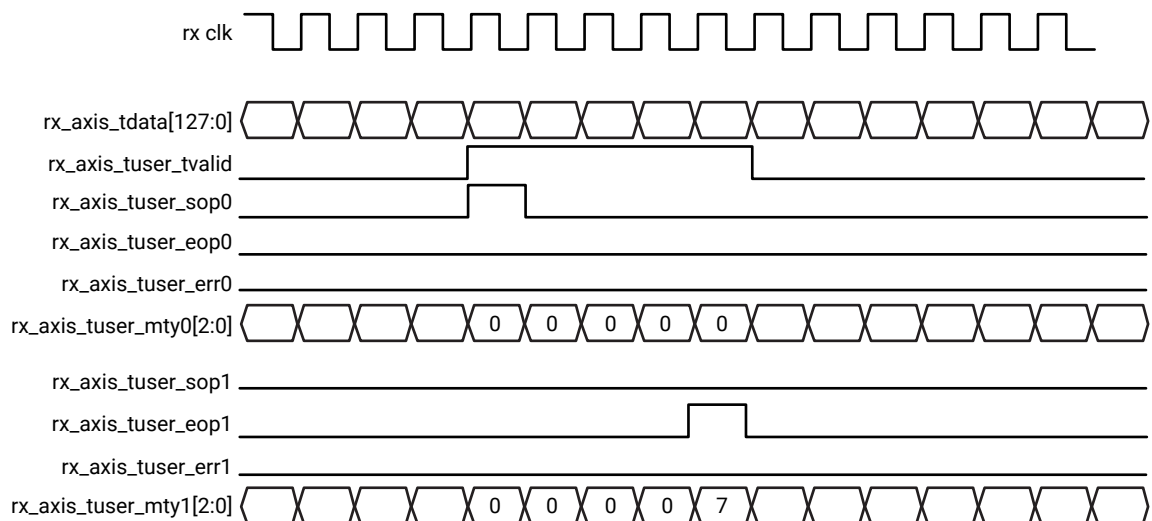
信号	I/O	时钟域	描述
rx_core_clk	输入		所有 RX AXI 信号均以此时钟为参考。
rx_axis_tdata[127:0]	输出	rx_core_clk	AXI4-Stream 数据，可连接到用户逻辑。
rx_axis_tuser_tvalid	输出	rx_core_clk	AXI4-Stream 数据有效。此信号为 1 时，表示 RX AXI 数据总线上的数据有效。
rx_axis_tuser_sop0 rx_axis_tuser_sop1	输出	rx_core_clk	此信号断言有效时，表示接收到的以太网帧起始。
rx_axis_tuser_eop0 rx_axis_tuser_eop1	输出	rx_core_clk	此信号断言有效时，表示接收到的以太网帧结束。2 位 - 每个分段 1 位。
rx_axis_tuser_err0 rx_axis_tuser_err1	输出	rx_core_clk	RX AXI 错误指示信号。 · 1 指示接收到的包无效。 · 0 指示接收到的包有效。2 位 - 每个分段 1 位。
rx_axis_tuser_mty0[2:0] rx_axis_tuser_mty1[2:0]	输出	rx_core_clk	此总线用于指示对应当前包的最近一次传输的 rx_axis_tdata 总线中为空或无效的字节数。此总线仅在 rx_axis_tuser_ena 和 rx_axis_tuser_eop 均为 1 的周期内有效。 2 位 - 每个分段 1 位。
rx_axis_tuser_ena0 rx_axis_tuser_ena1	输出	rx_core_clk	针对每个分段启用 AXI4-Stream 接收。此信号断言有效时，表示关联分段的数据有效。

注释：在 128 位跨接包接口中，TX AXIS 接口通常与 tx_out_clk 保持同步，RX AXIS 接口与 rx_clk_out 保持同步。

正常帧接收

正常入站帧传输的时序如下所示。客户端必须随时准备好接受数据，因为核内缓冲不足以允许接收客户端有时延。当帧接收开始时，数据在连续的时钟周期内传输到接收客户端。在帧接收期间，rx_axis_tuser_tvalid 断言有效，以指示有效的帧数据正通过 rx_axis_tdata 传输到客户端。所有 rx_axis_tuser_mty 位均设为 0，表示整个帧中的所有字节始终有效，但在 rx_axis_tuser_eop 断言有效时该帧进行最终传输期间除外。在帧的最终数据传输期间，rx_axis_tuser_mty 位指示帧的最终有效字节。

图 4: RX 波形图



X16337-080621

如需了解有关上方波形图中所示信号的描述，请参阅以下小节。

- rx_clk: rx_clk_out 信号应用作所有 RX AXI4-Stream 信号的时钟参考。所有 RX AXI 信号都与此时钟的上升沿对齐。
- rx_axis_tdata[127:0]: 此总线提供与所接收的以太网帧相对应的包导向数据。数据通过 rx_clk_out 信号计时。下图显示了如何将以太网帧的结束位置映射到 RX AXI4-Stream 接口的位元位置。请注意 ENAO 和 ENA1 信号的位置与 RX AXI4-Stream 总线的位元位置之间的对应关系。此映射适用于 128 位 AXI4-Stream 总线。

下图显示了含不间断连续传输的正常接收周期。124 字节的包起始于分段 0，结束于分段 1。其后是一个 99 字节的包，此包起始于分段 0 中的下一个时钟周期，结束于分段 0。

图 5: 含不间断连续传输的 RX 波形

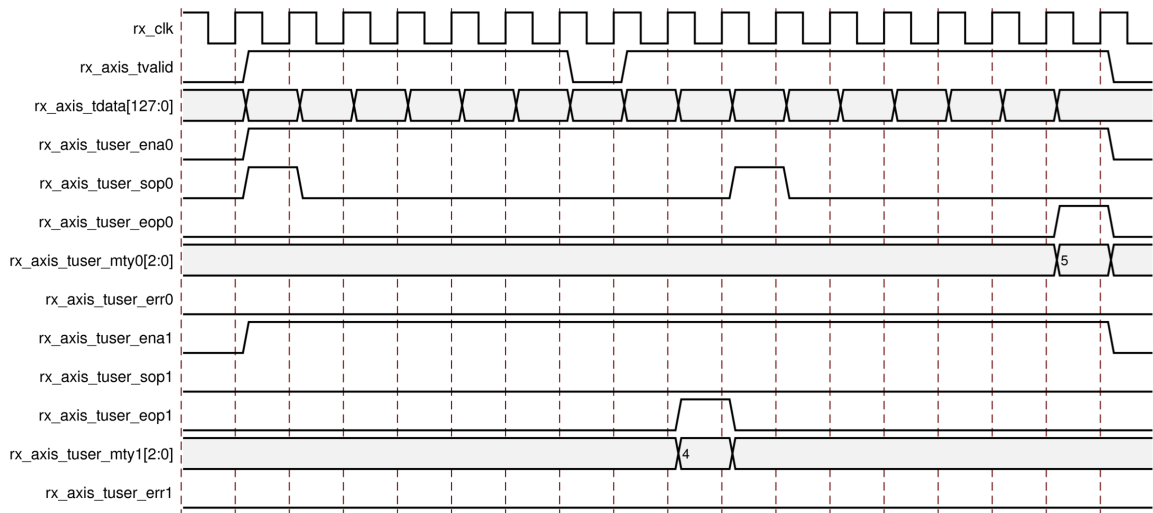
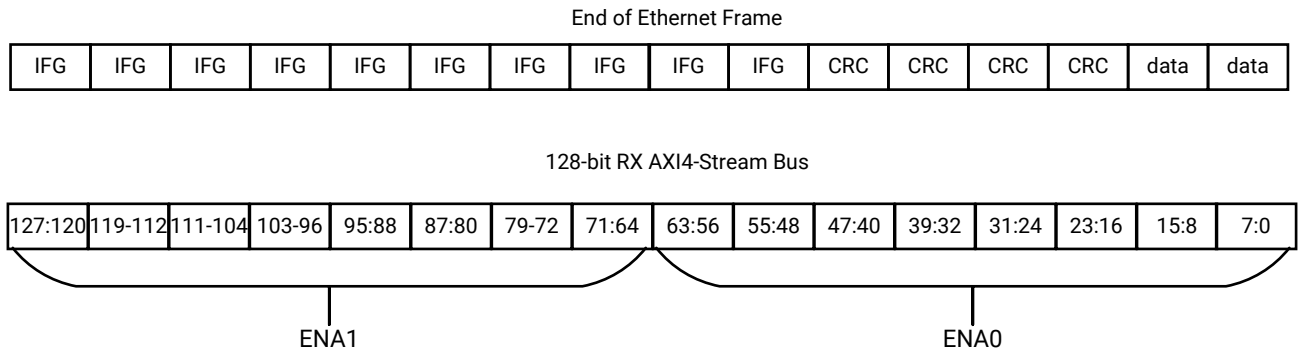


图 6: RX 映射



X16335-080621

- rx_axis_tuser_tvalid: 此信号断言有效时，表示 RX AXI 总线上的数据有效。
- rx_axis_tuser_ena0 和 rx_axi_tuser_ena1: 此信号断言有效时，表示关联分段的数据有效。
- rx_axis_tuser_sop0 和 rx_axis_tuser_sop1: 此信号指示 RX AXI 总线上以太网帧的起始位置。每个分段对应一个 SOP 信号。

- rx_axis_tuser_eop0 和 rx_axis_tuser_eop1：此信号指示 RX AXI 总线上以太网帧的结束位置。每个分段对应一个 EOP 信号。
- rx_axis_tuser_err0 和 rx_axis_tuser_err1：此信号断言有效时，表示接收的帧中含有错误。当 ena 和 tvalid 断言有效时，它在 EOP 周期内有效。每个分段对应一个错误信号。

错误类型可能包括：

- 发生了 FCS 错误
- 长度超出有效范围
- 接收数据包期间收到 64B/66B 代码错误
- rx_axis_tuser_mty0[2:0] 和 rx_axis_tuser_mty1[2:0]：与 TX AXI 接口中一样，mty 信号指示在接收数据包的最后一个周期（EOP 周期）内，当前周期中未使用（为空）的字节数。每个 RX 分段对应一个 mty 信号。

AXI4-Stream 发射接口

下图显示了 AXI4-Stream 发射接口信号。

表 7：AXI4-Stream 发射接口信号

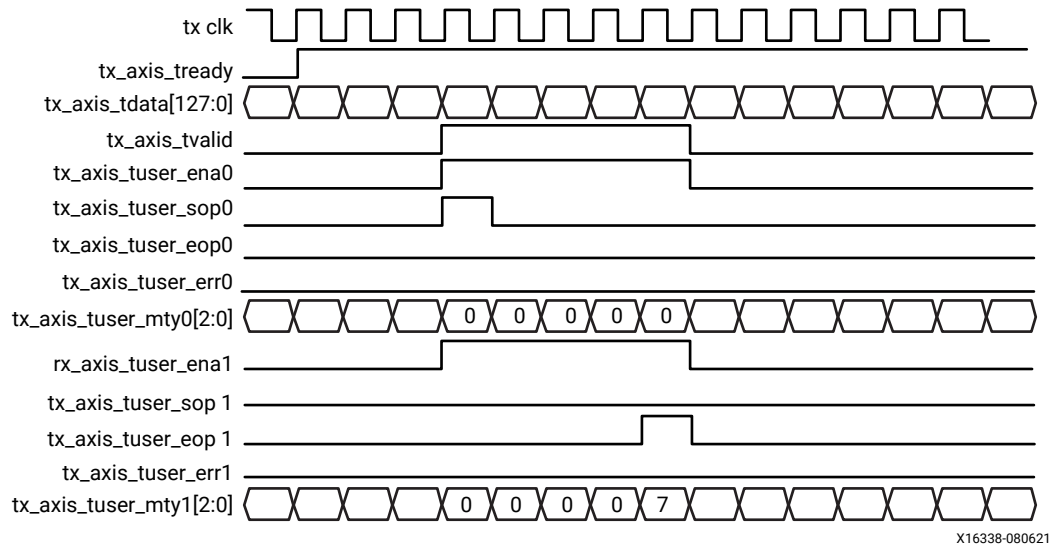
信号	I/O	时钟域	描述
tx_clk_out	输出		AXI 发射时钟。所有 TX 信号均以此时钟为参考。
tx_axis_tready	输出	tx_clk_out	当设为高电平有效 (High) 时，此信号指示 TX AXI 接口已准备好接受数据。当 tx_axis_tready 转为低电平有效 (Low) 时，您必须立即响应，即停止数据传输。
tx_axis_tdata[127:0]	输入	tx_clk_out	AXI4-Stream 发射数据（128 位接口）。TX AXI 数据总线用于接收用户提供的包数据。
tx_axis_tvalid	输入	tx_clk_out	AXI4-Stream 数据有效输入。仅当此信号为 1 时，数据传输才会完成。
tx_axis_tuser_ena0 tx_axis_tuser_ena1	输入	tx_clk_out	对应 TX AXI 总线传输的使能信号。此信号为“High”（高电平）即可启用向 TX 传输数据。
tx_axis_tuser_sop0 tx_axis_tuser_sop1	输入	tx_clk_out	AXI4-Stream 信号，指示以太网包起始。每个段均有 1 个包起始 (SOP) 信号。
tx_axis_tuser_eop0 tx_axis_tuser_eop1	输入	tx_clk_out	AXI4-Stream 信号，指示以太网包结束。每个段均有 1 个包结束 (EOP) 信号。
tx_axis_tuser_err0 tx_axis_tuser_err1	输入	tx_clk_out	此信号采样为 1 时用于指示数据包含有错误，针对包的所有其它传输则采样为 0。仅在 tx_enain 和 tx_eopin 均采样为 1 的周期内才对此信号进行采样。当此信号采样为 1 时，最后一个代码字将被替换为 802.3 错误码控制字，以保证伙伴器件可接收到存在错误的包。如果在包的输入端此信号设置为 1，则禁用 FCS 检查和报告（仅针对该数据包）。每个段均有 1 个信号。
tx_axis_tuser_mty0[2:0] tx_axis_tuser_mty1[2:0]	输入	tx_clk_out	发射空数据。此总线用于指示对应当前包的最近一次传输的 tx_datain 总线中为空或无效的字节数。仅在 tx_axis_valid 和 tx_axis_user_eopin 均采样为 1 的周期内才对此总线进行采样。

同步 TX AXI 总线接口可接受包导向数据。所有信号均与 tx_clk_out 端口的上升沿保持相对同步。

AXI4-Stream 发射接口包含 2 个段，每个段宽度均为 64 位（8 字节）。此分段式方法也称为跨接式 AXI4-Stream 方法，能够提升效率，以使数据包能够在给定的分段或周期内起始和结束。

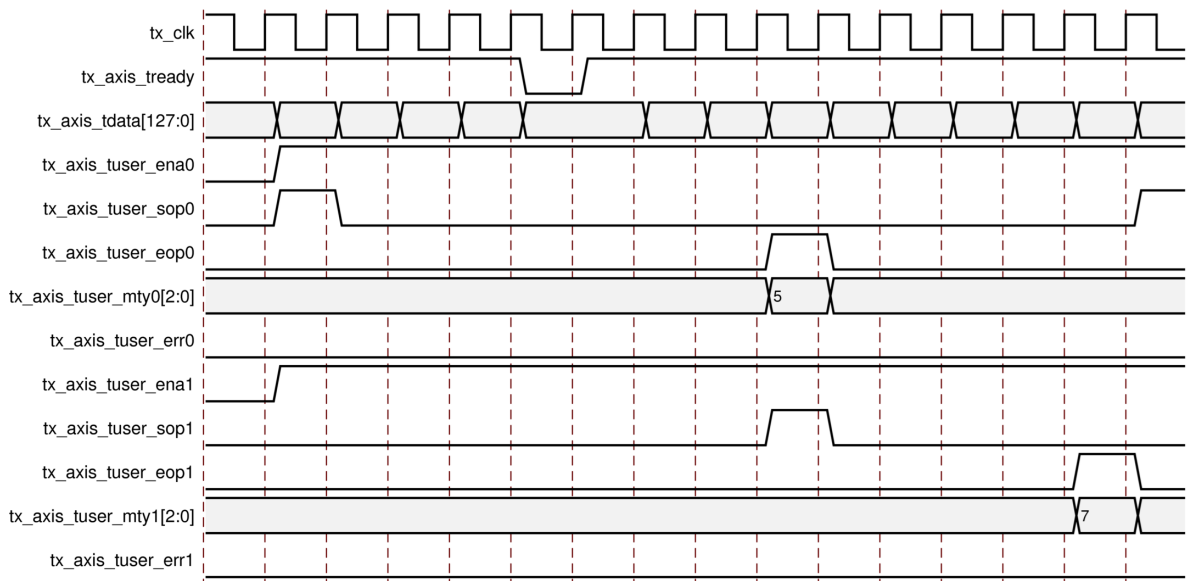
以下波形中显示了正常的发射周期。这些波形展示了在位宽为 128 位的 AXI4-Stream 接口上传输 73 字节的包的过程。其中显示了 2 个段，编号为 0 和 1。

图 7: TX 波形图



下图显示了含不间断连续传输的正常发射周期。115 字节的包起始于分段 0，结束于分段 0。紧随其后是一个 81 字节的包，此包起始于分段 1 中的同一个时钟周期，结束于分段 1。

图 8: 含不间断连续传输的 TX 波形



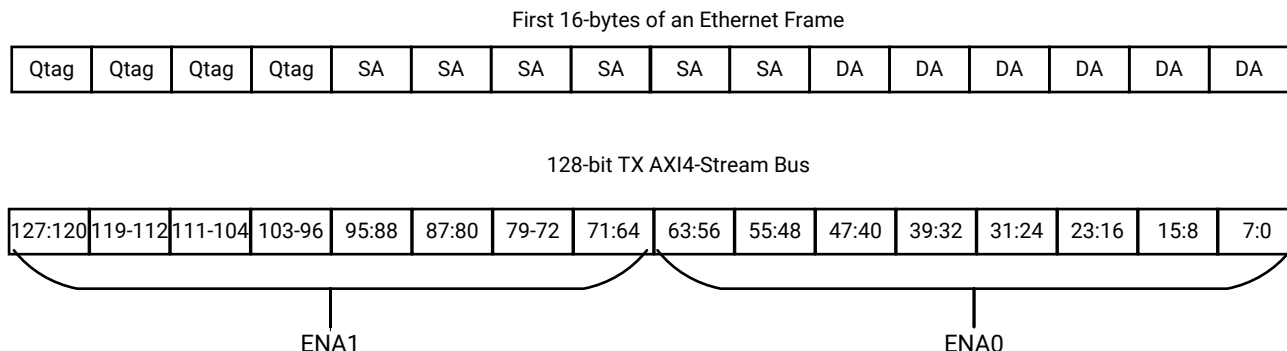
以下小节中对 [图 7: TX 波形图](#) 和 [图 8: 含不间断连续传输的 TX 波形](#) 波形中所示信号进行了描述。

- tx_clk: 这是子系统的信号 tx_clk_out 输出。所有 TX AXI 信号均以此时钟为参考。对于 50G 运行速度，此时钟的频率通常为 390.625 MHz，对于 40G 运行速度则通常为 312.5 MHz。
- tx_axis_tready: 此信号断言有效时，表示 TX AXI4-Stream 接口能够接受数据。当 tx_axi_tready 变为低电平有效时，必须立即停止发送数据，否则 TX AXI4-Stream 接口将不接受数据。
- tx_axis_tdata[127:0]: 这是要发射的帧的总线。

下图演示了当 tx_axis_tuser_sop0 = 1 时，如何将以太网帧起始位置映射到 TX AXI4-Stream 接口的位元位置。请注意与 TX AXI4-Stream 总线的位元位置对应的 ENAO 和 ENA1 信号的位置。

此映射对应 128 位 AXI4-Stream 总线。

图 9：TX 映射



X16336-080621

- tx_axis_tvalid: 当设为“High”时，此信号指示 TX AXI 总线上存在有效数据。
- tx_axis_tuser_ena0 和 tx_axis_tuser_ena1: 这些信号断言有效时，支持通过 TX 总线来传输数据。数据传输必须通过 tvalid 信号加以确认，才能执行传输。

针对每个 AXI 信号段都存在 1 个使能信号。

- tx_axis_tuser_sop0 和 tx_axis_tuser_sop1: 这些信号表示该周期内以太网帧的起始位置。每个总线周期内仅允许 1 个 SOP。针对每个 AXI 信号段都存在 1 个单独的 SOP 信号。
- tx_axis_tuser_eop0 和 tx_axis_tuser_eop1: 这些信号表示该周期内以太网帧的结束位置。每个总线周期内仅允许 1 个 EOP。针对每个 AXI 信号段都存在 1 个单独的 EOP 信号。
- tx_axis_tuser_mty0[2:0] 和 tx_axis_tuser_mty1[2:0]: 这些信号表示对应信号段的哪些字节处于未使用（“为空”）状态。如果 tx_mtyin 值为 0x0，则不存在空字节通道，换言之，数据总线的所有位均有效。例如，如果 tx_axis_tuser_mty0[2:0] = 2，那么该信号段的最后 2 个字节不包含数据，且将被忽略。在包传输的最后一个周期（EOP 周期）内，mty 只能包含非零 (0) 值。
- tx_unfout: 在前述 TX 波形时序图中的波形上未显示 tx_unfout 输出指示符。当此信号设为“High”时，表示没有足够的数据传输，以太网接口将下溢。不允许出现此状况。您必须确保只要 tx_axis_tready 为“High”，就必须存在数据传输，直至达到以太网帧结束为止。

注释：当此信号采样为 1 时，您需要应用 tx_reset/sys_reset 以便使该核从下溢问题恢复。tx_reset 仅复位 TX 路径，sys_reset 则会恢复整个系统。

- tx_axis_user_err0 和 tx_axis_user_err1: 此信号采样为 1 时用于指示数据包含有错误，针对包的所有其它传输则采样为 0。仅在 tx_axis_tuser_ena 和 tx_axis_tuser_eop 均采样为 1 的周期内才对此信号进行采样。当此信号采样为 1 时，最后一个代码字将被替换为 802.3 错误码控制字，以保证伙伴器件可接收到存在错误的数据包。如果在包的输入端此信号设置为 1，则禁用 FCS 检查和报告（仅针对该数据包）。每个段均有 1 个 tx_axis_user_err 信号。

AXI4-Stream 接口 - 256 位包接口

40G/50G IP 子系统针对 40 Gb/s 运行提供了 256 位 AXI4-Stream 包接口选项。

AXI4-Stream 时钟和复位

表 8: AXI4-Stream 接口 - 时钟/复位信号

名称	I/O	描述
clk	输入	AXI4-Stream 时钟。40G/50G High Speed Ethernet Subsystem 与用户端逻辑之间的所有信号均同步到此信号的上升沿。
dclk	输入	此时钟必须为稳定且便于使用的时钟，例如，75 MHz。请参阅最新收发器指南以获取最新信息。
rx_reset	输入	RX 电路的复位。此信号为高电平有效（1 = 复位），并且必须保持高电平，直至 clk 达到稳定状态为止。40G/50G High Speed Ethernet Subsystem 负责处理将 rx_reset 输入同步到核中相应的时钟域的操作。这是同步复位。
refclk_n0	输入	对应收发器的差分参考时钟 (N)。
refclk_p0	输入	对应收发器的差分参考时钟 (P)。
tx_reset	输入	TX 电路的复位。此信号为高电平有效（1 = 复位），并且必须保持高电平，直至 clk 达到稳定状态为止。40G/50G High Speed Ethernet Subsystem 负责处理将 tx_reset 输入同步到核中相应的时钟域的操作。这是同步复位。

AXI4-Stream 发射接口 - 256 位

下表显示了 AXI4-Stream 发射接口信号。

表 9: AXI4-Stream 发射接口

名称	I/O	时钟	描述
tx_axis_tready	输出	tx_clk_out	AXI4-Stream 确认信号，用于指示开始数据传输
tx_axis_tvalid	输入	tx_clk_out	AXI4-Stream 数据有效输入
tx_axis_tdata[255:0]	输入	tx_clk_out	AXI4-Stream 数据
tx_axis_tuser	输入	tx_clk_out	AXI4-Stream 用户边带接口。 1 指示包无效。0 指示包有效。
tx_axis_tlast	输入	tx_clk_out	AXI4-Stream 信号，指示以太网包结束
tx_axis_tkeep[31:0]	输入	tx_clk_out	AXI4-Stream 数据控制

数据通道映射 - TX

对于发射数据 tx_axis_tdata[255:0]，端口按逻辑分割为通道 0 到通道 31。请参阅下表。

表 10: tx_axis_tdata 通道

通道/tx_axis_tkeep	tx_axis_tdata[255:0] 位
0	7:0
1	15:8
2	23:16
3	31:24
4	39:32
5	47:40
6	55:48
7	63:56

表 10: tx_axis_tdata 通道 (续)

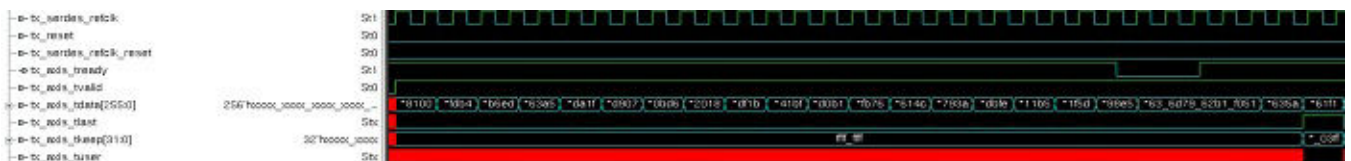
通道/tx_axis_tkeep	tx_axis_tdata[255:0] 位
8	71:64
9	79:72
10	87:80
11	95:88
12	103:96
13	111:110
14	119:112
15	127:120
16	135:128
17	143:136
18	151:144
19	159:152
20	167:160
21	175:168
22	183:176
23	191:184
24	199:192
25	207:200
26	215:208
27	223:216
28	231:224
29	239:232
30	247:240
31	255:248

正常发射

正常帧传输的时序如下图所示。当客户端发射帧时，它会断言 tx_axis_tvalid 信号有效并将 tx_axis_tdata 和 tx_axis_tkeep 中的数据和控制布局在同一个时钟周期内。核接受此数据时（通过 tx_axis_tready 断言有效来指示），客户端必须提供下一个周期的数据。如果核未断言 tx_axis_tready 有效，则客户端必须保持当前有效数据值，直至其断言有效为止。通过保持 1 个周期的 tx_axis_tlast asserted 向核指示包结束。正确设置 tx_axis_tkeep 的位，以指示最终数据传输中的有效字节数。通过同时断言 tx_axis_tuser 有效来指示数据包无效。

在 tx_axis_tlast 断言无效后，所有数据与控制都被视为无效，直至下一次 tx_axis_tvalid 断言有效为止。

图 10: 正常帧发射 - 256 位 AXI4-Stream



连续传输

在 AXI4-Stream 发射接口上之所以能够连续传输数据，原因在于 tx_axis_tvalid 信号能够连续保持高电平，并且包边界只能通过断言 tx_axis_tlast 有效（表示以太网包结束）来定义。不过，核可以根据需要通过断言 tx_axis_tready 确认信号无效来对客户端数据进行节流。请参阅下图。

在核断言 tx_axis_tready 确认信号无效时，客户端数据逻辑可以用有效的数据来更新 AXI4-Stream 接口。不过，当 valid 断言有效并且新数据置于 AXI4-Stream 上时，它应保留不动，直到核断言 tx_axis_tready 有效为止。

图 11：连续传输 - 256 位 AXI4-Stream



中止发射

客户端接口数据包传输中止称为欠载 (underrun)。如果 AXI 发射客户端接口中的 FIFO 在帧完成之前清空，则可能发生这种情况。

通过以下任一方法即可向核指明发生发射中止。

- 显式错误，由于 asserting tx_axis_tuser 为高电平且 tx_axis_tlast 为高电平而导致帧传输中止。
- 隐式欠载，由于 tx_axis_tvalid 断言无效且 tx_axis_tlast 未断言有效而导致帧传输中止。

AXI4-Stream 接收接口 - 256 位

表 11：AXI4-Stream 接收接口

名称	I/O	时钟	描述
rx_axis_tdata[255:0]	输出	rx_core_clk	AXI4-Stream 数据到用户逻辑
rx_axis_tvalid	输出	rx_core_clk	AXI4-Stream 数据有效。当此信号设为 1 时，RX AXI 总线上存在有效数据。
rx_axis_tuser	输出	rx_core_clk	AXI4-Stream 用户边带接口。 · 1 指示接收到的包无效。 · 0 指示接收到的包有效。
rx_axis_tlast	输出	rx_core_clk	AXI4-Stream 信号，指示包结束
rx_axis_tkeep[31:0]	输出	rx_core_clk	AXI4-Stream 数据控制，连接到上层。

注释：在 256b 包接口模式下，TX 和 RX AXI4-Stream 接口同步到单一输入时钟，rx_core_clk 通常由 tx_clk_out 驱动。

数据通道映射 - RX

对于接收数据 rx_axis_tdata[63:0]，端口按逻辑划分为通道 0 到通道 7。请参阅下表。

表 12：数据通道映射

Lane/rx_axis_tkeep	rx_axis_tdata[255:0] 位
0	7:0

表 12: 数据通道映射 (续)

Lane/rx_axis_tkeep	rx_axis_tdata[255:0] 位
1	15:8
2	23:16
3	31:24
4	39:32
5	47:40
6	55:48
7	63:56
8	71:64
9	79:72
10	87:80
11	95:88
12	103:96
13	111:110
14	119:112
15	127:120
16	135:128
17	143:136
18	151:144
19	159:152
20	167:160
21	175:168
22	183:176
23	191:184
24	199:192
25	207:200
26	215:208
27	223:216
28	231:224
29	239:232
30	247:240
31	255:248

正常帧接收

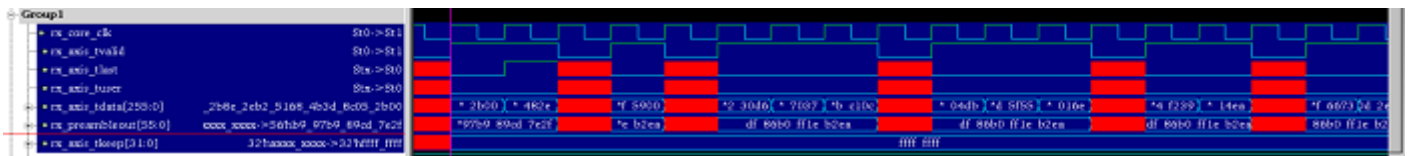
正常入站帧传输的时序如下图所示。客户端必须准备好随时接受数据；由于核中没有缓冲机制，因此接收客户端不允许出现时延。

帧接收期间，rx_axis_tvalid 断言有效以指示正在 rx_axis_tdata 上将有效帧数据传输到客户端。所有 rx_axis_tkeep 位均设为 1，表示整个帧中的所有字节始终有效，但在 rx_axis_tlast 断言有效时该帧进行最终传输期间除外。在帧数据的最终传输期间，rx_axis_tkeep 位使用上述映射来指示帧的最终有效字节。

最终传输的有效字节始终来自于 rx_axis_tdata[7:0] (rx_axis_tkeep[0])，因为以太网帧数据是连续的并且先接收最低有效字节。

仅在完成所有帧校验后，随传输的最终字节一起，rx_axis_tlast 断言有效并且 rx_axis_tuser 断言无效。这发生在接收到帧校验序列 (FCS) 字段后。核断言 rx_axis_tuser 信号有效以指示已成功接收到帧并且客户应对帧进行分析。这也表示包结束，以 rx_axis_tlast 断言有效并保持 1 个周期来表示。

图 12: 正常帧接收 - 256 位非分段式 AXI4-Stream



帧接收时出错

下图显示了帧接收不成功的情况（例如，超短帧或含错误 FCS 的帧）。在此情况下，接收到的帧为坏帧，并且在此帧结束时会对客户端断言 rx_axis_tuser 信号有效。随后，客户端负责丢弃已为此帧传输的数据。

以下条件会导致断言 rx_axis_tlast 有效并且 rx_axis_tuser = 1（表示坏帧）。

- 发生 FCS 错误
- 数据包长度小于 64 字节（大小过小或者碎片帧）
- 接收到的帧的长度超过已编程的最大传输单元 (MTU) 大小。
- 接收到的任何控制帧的长度都并非恰好为最小帧长度。
- XLGMII 数据串流包含错误代码。

图 13: 帧接收时出错 - 256 位非分段式 AXI4-Stream



TX 路径控制信号和状态端口

下表描述了其它状态端口和控制端口。

表 13: TX 路径控制信号和状态端口

名称	I/O	时钟域	描述
ctl_rate_mode	输入	static	此信号导致 IP 核在 50G 运行 (0) 和 40G 运行 (1) 之间切换。请注意，必须针对所选模式纠正时钟频率。
ctl_tx_enable	输入	clk	TX 使能 (启用)。此信号采样为 1 时用于启用数据发射。当此信号采样为 0 时，40G/50G High Speed Ethernet 子系统仅发射空闲信号。仅当数据发射到的接收器 (即其它器件中的接收器) 完全对齐并且已准备好接收数据 (即，其它器件当前未发送远端故障状况) 后，此输入才应设置为 1。否则，可能发生数据丢失。如果发射数据包时此信号设置为 0，那么当前数据包发射完成后，40G/50G High Speed Ethernet 子系统将停止发射任何其它数据包。

表 13: TX 路径控制信号和状态端口 (续)

名称	I/O	时钟域	描述
ctl_tx_send_rfi	输入	clk	发射远端故障指示 (RFI) 代码字。如果此输入采样为 1, 那么 TX 路径仅发射远端故障 (Remote Fault) 代码字。此输入应设置为 1, 直至 RX 路径已完全对齐并且已准备好接受来自链路伙伴的数据为止。
ctl_tx_send_lfi	输入	clk	发射本地故障指示 (LFI) 代码字。优先于 RFI。
ctl_tx_send_idle	输入	clk	发射空闲代码字。如果此输入采样为 1, 那么 TX 路径仅发射空闲 (Idle) 代码字。当伙伴器件正在发送远端故障指示 (RFI) 代码字时, 此输入应设置为 1。
ctl_tx_fcs_ins_enable	输入	clk	启用由 TX 核执行 FCS 插入。如果此位设置为 0, 40G/50G High Speed Ethernet 子系统不会将 FCS 添加到包中。如果此位设置为 1, 则 40G/50G High Speed Ethernet Subsystem 会计算 FCS 并将其添加到包中。在不同数据包之间不得动态更改此输入。
ctl_tx_ignore_fcs	输入	clk	启用由 TX 核在 AXI4-Stream 接口上执行 FCS 纠错。仅当 ctl_tx_fcs_ins_enable 为低电平时, 此输入才有效。如果此输入为低电平, 并且发射的数据包中含有无效 FCS, 则不会将其作为有效包丢弃。如果此输入为高电平, 则含无效 FCS 的包将被作为有效包丢弃。 在 stat_tx_bad_fcs 和 stomped_fcs 信号上会标记此错误, 并且此数据包将作为已接收的包进行发射。 注释: 报告的统计数据中显示不含 FCS 错误。
ctl_tx_vl_length_minus1[15:0]	输入	static	PCS 通道标记间的字数减 1。根据 IEEE 802.3-2015 标准中的定义, 默认值应设置为 16,383。仅当对应的复位输入断言有效时, 才应更改此输入。 注释: 在 50G 核配置中启用 RS-FEC 时, 此值将设置为 20479。
ctl_tx_vl_marker_id[VL_LANES-1:0][63:0]	输入	static	这些输入为每个 PCS 通道设置 PCS 通道标记。如需了解 802.3 默认值, 请参阅 IEEE 802.3-2015 标准《IEEE 以太网标准》(IEEE 802.3-2015 标准)。仅当对应的复位输入断言有效时, 才应更改此输入。
stat_tx_local_fault	输出	clk	值为 1 表示发射编码器状态机当前处于 TX_INIT 状态。此输出对电平敏感。
ctl_tx_custom_preamble_enable	输入	tx_clk	此信号断言有效时, 会将 rx_serdes_clk 上数据包的前 64 位作为自定义前导码来处理, 而不是插入标准前导码。 此信号断言有效时, 支持使用 tx_preamblein 作为自定义前导码, 而不是插入标准前导码。
tx_preamblein[55:0]	输入	tx_clk	当 ctl_tx_custom_preamble_enable 信号断言有效时, 此总线表示自定义前导码。它应在数据包的第一个周期 (包起始) 断言有效。
stat_tx_underflow_err ¹	输出	tx_clk	TX FIFO 下溢
stat_tx_overflow_err ¹	输出	tx_clk	TX FIFO 上溢

注释:

1. 这些信号仅在 256 位非分段式 AXI4-Stream 变体中可用。

RX 路径控制信号和状态端口

下表描述了其它状态端口和控制端口。

表 14: RX 路径控制信号和状态信号

名称	I/O	时钟域	描述
ctl_rate_mode	输入	static	此信号导致 IP 核在 50G 运行 (0) 和 40G 运行 (1) 之间切换。请注意，必须针对选中的模式纠正时钟频率。
ctl_rx_enable	输入	rx_serdes_clk	RX 使能 (启用)。在正常操作期间，此输入必须设置为 1。当此输入设置为 0 时，RX 完成接收当前包 (如果有) 之后，就会停止接收包并阻止 PCS 继续对传入数据进行解码。在此模式下，不报告任何统计数据，并且 AXI4-Stream 接口处于空闲状态。
ctl_rx_check_preamble	输入	rx_serdes_clk	此输入断言有效时，会使 Ethernet MAC 检查接收到的帧的前导码。
ctl_rx_check_sfd	输入	rx_serdes_clk	此输入断言有效时，会使 Ethernet MAC 检查接收到的帧的帧起始定界符。
ctl_rx_force_resync	输入	rx_serdes_clk	RX 强制再同步输入。此信号用于强制 RX 路径进行复位、再同步和重新对齐。值为 1 即强制执行复位操作。值为 0 允许正常操作。 注释： 此输入通常应为低电平 (Low) 并且仅限用于强制重新对齐时才应进行脉冲 (单周期最小脉冲)。CTL_RX_FORCE_RESYNC 会重新启动同步状态机，但是不会复位 GT 逻辑。在大多数情况下，如果发生 RX 故障，则需要复位 GT RX。
ctl_rx_delete_fcs	输入	rx_serdes_clk	启用于 RX 核执行 FCS 移除。如果将此位设置为 0，则 40G/50G High Speed Ethernet 子系统不会移除传入包的 FCS。如果将此位设置为 1，则 40G/50G High Speed Ethernet Subsystem 会删除所接收包的 FCS。对于长度为 8 字节的包，不删除 FCS。仅当对应的复位输入断言有效时，才应更改此输入。
ctl_rx_ignore_fcs	输入	rx_serdes_clk	允许 RX 核在 AXI4-Stream 接口上执行 FCS 错误检查。如果此位设置为 0，对于接收到的含 FCS 错误的包，将随在上一次传输期间断言有效 (rx_eopout 和 rx_enaout 均采样为 1) 的 rx_errout 管脚一起发送。如果将此位设置为 1，则 40G/50G High Speed Ethernet Subsystem 不会标记 AXI4-Stream 接口上的 FCS 错误。 注释： 报告的统计数据中显示此数据包有效。但是，stat_rx_bad_fcs 信号会报告错误。
ctl_rx_max_packet_len[14:0]	输入	rx_serdes_clk	长度超过该值的任何包都将被视为大小过大。如果包的大小大于该值，则该包将被截位至该值，并且 rx_errout 信号随 rx_eopout 信号一起断言有效。小于 16 字节的包将被丢弃。该总线允许的值范围为 64 到 16,383。 ctl_rx_max_packet_len[14] 保留并且必须设置为 0。
ctl_rx_min_packet_len[7:0]	输入	rx_serdes_clk	长度小于该值的任何包都将被视为大小过小。如果包的大小小于该值，则在 rx_eopout 断言有效的周期内，rx_errout 信号将断言有效。小于 16 字节的包将被丢弃。 注释： 该值应大于或等于 64B。
ctl_rx_vl_length_minus1[15:0]	输入	rx_serdes_clk	PCS 通道标记间的字数减 1。根据 IEEE 802.3-2015 标准中的定义，默认值应设置为 16,383。仅当对应的复位输入断言有效时，才应更改此输入。 注释： 在 50G 核配置中启用 RS-FEC 时，此值将设置为 20479。

表 14: RX 路径控制信号和状态信号 (续)

名称	I/O	时钟域	描述
ctl_rx_vl_marker_id[VL_LANES-1:0][63:0]	输入	rx_serdes_clk	这些输入为每个 PCS 通道设置 PCS 通道标记。这些输入应设置为 IEEE 802.3-2015 标准中定义的值。如需了解 IEEE 802.3 默认值, 请参阅《IEEE 以太网标准》(IEEE 802.3-2015 标准) 第 5.3 节。仅当对应的复位输入断言有效时, 才应更改此输入。
stat_rx_framing_err_[VL_LANES-1:0][3:0]	输出	rx_clk_out	RX 同步报头位成帧错误。每个 PCS 通道都有 1 个 4 位总线, 用于指示针对该 PCS 通道接收到的同步报头错误的数量。仅当对应的 stat_rx_framing_err_valid_[VL_LANES-1:0] 为 1 时, 该总线的值才有效。这些总线上的值可随时更新, 并且这些值旨在作为递增值以供同步报头错误计数器使用。
stat_rx_framing_err_valid_[VL_LANES-1:0]	输出	rx_clk_out	stat_rx_framing_err_[VL_LANES-1:0] 的有效指示符。当这些输出之一采样为 1 时, 对应 stat_rx_framing_err_[VL_LANES-1:0] 上的值有效。
stat_rx_local_fault	输出	rx_clk_out	当 stat_rx_internal_local_fault 或 stat_rx_received_local_fault 断言有效时, 此输出为高电平 (High)。此输出对电平敏感。
stat_rx_synced[VL_LANES-1:0]	输出	rx_clk_out	字边界已同步。这些信号用于指示 PCS 通道是否已完成字边界同步。值为 1 表示对应 PCS 通道已实现字边界同步, 并且已接收到 PCS 通道标记。 对应于第 82.3 条中所定义的管理数据输入/输出 (MDIO) 寄存器位 3.52.7:0 和 3.53.11:0。此输出对电平敏感。
stat_rx_synced_err[VL_LANES-1:0]	输出	rx_clk_out	字边界同步错误。这些信号用于指示在相应的 PCS 通道内进行字边界同步期间, 是否发生错误。值为 1 表示由于同步报头成帧位错误或者从未收到 PCS 通道标记, 对应 PCS 通道已丢失字边界同步。此输出对电平敏感。
stat_rx_mf_len_err[VL_LANES-1:0]	输出	rx_clk_out	PCS 通道标记长度错误。这些信号用于指示相应通道内是否发生了 PCS 通道标记不匹配 (即, 接收到的 PCS 通道标记间隔字数并非 ctl_rx_vl_length_minus1)。值为 1 表示对应通道接收 PCS 通道标记的间隔错误。 它将保持高电平, 直至消除错误状况为止。
stat_rx_mf_repeat_err[VL_LANES-1:0]	输出	rx_clk_out	PCS 通道标记连续错误。这些信号用于指示相应通道内是否发生了 4 个连续 PCS 通道标记错误。值为 1 表示对应通道内存在错误。此输出将保持高电平 (High), 直至消除错误状况为止。
stat_rx_mf_err[VL_LANES-1:0]	输出	rx_clk_out	PCS 通道标记字错误。这些信号用于表示在相应通道内检测到构成的 PCS 通道标记字错误。值为 1 表示发生了错误。此输出将脉冲 1 个时钟周期, 以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_aligned	输出	rx_clk_out	所有 PCS 通道均已对齐/去歪斜。此信号表示是否所有 PCS 通道均已对齐并去歪斜。值为 1 表示所有 PCS 通道均已对齐并去歪斜。当此信号为 1 时, 表示 RX 路径已对齐并且可接收包数据。当此信号为 0 时, 表示存在本地故障状况。这也对应于第 82.3 条中所定义的 MDIO 寄存器位 3.50.12。此输出对电平敏感。
stat_rx_status	输出	rx_clk_out	PCS 状态。值为 1 表示 PCS 已对齐且未处于 hi_ber 状态。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.32.12。此输出对电平敏感。
stat_rx_block_lock[VL_LANES-1:0]	输出	rx_clk_out	每个 PCS 通道的块锁定状态。值为 1 表示对应通道已达到第 82 条中定义的块锁定状态。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.50.7:0 和 3.51.11:0。此输出对电平敏感。

表 14: RX 路径控制信号和状态信号 (续)

名称	I/O	时钟域	描述
stat_rx_aligned_err	输出	rx_clk_out	通道对齐/去歪斜丢失。此信号表示 PCS 通道对齐期间发生错误或者 PCS 通道对齐已丢失。值为 1 表示发生了错误。此输出对电平敏感。
stat_rx_misaligned	输出	rx_clk_out	对齐错误。此信号表示通道对齐器并未在所有通道上都接收到期望的 PCS 通道标记。在所有通道上都至少接收到 1 个 PCS 通道标记并且至少接收到 1 个错误的通道标记后，此信号才会断言有效。此状况的发生比错误晚 1 个元帧。 如果从未正确接收到任何通道标记，则此信号不会断言有效。通道标记错误通过对应的 stat_rx_mf_err 信号来指示。 此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_remote_fault	输出	rx_clk_out	远端故障指示状态。如果该位采样为 1，则表示检测到远端故障状况。如果该位采样为 0，则表示不存在远端故障状况。此输出对电平敏感。
stat_rx_vl_number_[3:0] 1:0]	输出	rx_clk_out	总共有 VL_LANES 个独立的 stat_rx_vl_number[4 1:0] 总线。stat_rx_vl_number_# 指示在对应的物理通道上接收到的 PCS 通道。仅当 stat_rx_synced[VL_LANES-1:0] 的对应位为 1 时，该总线才有效。这些输出对电平敏感。
stat_rx_vl_demuxed[VL_LANES-1:0]	输出	rx_clk_out	已找到 PCS 通道标记。如果该总线的信号采样为 1，则表示接收器已对该 PCS 通道进行了正确的逆多路复用。这些输出对电平敏感。
stat_rx_bad_fcs[n:0]	输出	rx_clk_out	无效 FCS 指示符。该总线上的值用于指示接收到的包带有无效 FCS，但不带有含 stomp 标记的 FCS。含 stomp 标记的 FCS 定义为期望的有效 FCS 的按位反转。此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_stomped_fcs[n:0]	输出	rx_clk_out	含 stomp 标记的 FCS 指示符。该总线上的值用于指示接收到的包带有含 stomp 标记的 FCS。含 stomp 标记的 FCS 定义为期望的有效 FCS 的按位反转。此输出将脉冲 1 个时钟周期，以指示含 stomp 标记。脉冲可在不间断的连续周期内发生。
stat_rx_truncated	输出	rx_clk_out	包截位指示符。值为 1 表示当前进行中的包因长度超过 ctl_rx_max_packet_len[14:0] 而被截位。此输出将脉冲 1 个时钟周期，以指示截位状况。脉冲可在不间断的连续周期内发生。
stat_rx_internal_local_fault	输出	rx_clk_out	当由于以下任一操作导致生成内部本地故障时，此信号将转为高电平 (High)：测试模式生成、通道对齐错误或误码率过高。只要故障状况仍然存在，此信号就会保持高电平 (High)。
stat_rx_received_local_fault	输出	rx_clk_out	当接收到来自链路伙伴的本地故障字数足以触发 IEEE 故障状态机所指定的故障条件时，此信号将转为高电平 (High)。只要故障状况仍然存在，此信号就会保持高电平 (High)。
stat_rx_bip_err[VL_LANES-1:0]	输出	rx_clk_out	BIP8 错误指示符。非 0 值表示 BIP8 签名字节针对对应 PCS 通道存在错误。非 0 值将脉冲 1 个时钟周期。此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_hi_ber	输出	rx_clk_out	误码率 (BER) 过高指示符。设置为 1 时，BER 过高 (根据 IEEE 802.3-2015 标准的定义)。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.32.1。此输出对电平敏感。
ctl_rx_custom_preamble_enable	输入	rx_clk_out	此信号断言有效时，会导致在 rx_preambleout 上呈现前导码。

表 14: RX 路径控制信号和状态信号 (续)

名称	I/O	时钟域	描述
rx_preambleout[55:0]	输出	rx_clk	当 ctl_rx_custom_preamble_enable 信号断言有效时，该总线表示前导码字节。它在包的第一个周期内有效。

杂项状态端口和控制端口

下表描述了其它状态信号和控制信号。

表 15: 杂项状态端口和控制端口

名称	I/O	时钟域	描述
ctl_tx_ipg_value[3:0]	输入		此信号为可选。ctl_tx_ipg_value 用于定义在 rx_serdes_clk 包之间插入的目标平均最小包间间隔 (IPG, 以字节为单位)。典型值为 12。ctl_tx_ipg_value 还可编程为范围介于 0 到 7 之间的值, 但在此情况下, 它解释为表示“最小 IPG”, 因此仅插入终止 (Terminate) 代码字 IPG; 在此情况下从不添加空闲 (Idle), 因此发射随机大小的数据包时, 生成的 IPG 平均约为 4 字节。
stat_rx_got_signal_os	输出	rx_clk_out	Signal OS 指示。如果该位采样为 1, 则表示接收到“Signal OS”字。 注释: 在以太网网络中不应接收到 Signal OS
ctl_rx_process_lfi	输入	rx_clk_out	当此输入设为 1 时, RX 核预计应从 SerDes 传入本地故障 (LF) 控制代码并且会对其进行处理。设置为 0 时, RX 核会忽略从 SerDes 传入的 LF 控制代码。
ctl_rx_test_pattern	输入	rx_clk_out	针对 RX 核启用测试模式检查。值为 1 即表示根据第 82.2.17 条中的定义启用测试模式。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.42.2。检查扰码空闲模式。
ctl_tx_test_pattern	输入	clk	针对 TX 核启用测试模式生成。值为 1 即表示根据第 82.2.10 条中的定义启用测试模式。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.42.7。生成扰码空闲模式。
stat_rx_test_pattern_mismatch[3 2 1 0:0]	输出	rx_clk_out	测试模式不匹配数递增。任一周期内的非 0 值均表示 RX 核中在测试模式下发生的不匹配次数。仅当 ctl_rx_test_pattern 设置为 1 时, 此输出才有效。此输出可用于根据第 82.3 条的定义生成 MDIO 寄存器 3.43.15:0。此输出将脉冲 1 个时钟周期。

统计数据接口端口

以下 2 表分别描述了 RX 路径端口和 TX 路径端口。

表 16: 统计数据接口 - RX 路径信号

名称	I/O	时钟域	描述
stat_rx_total_bytes[6 5 3 2:0]	输出	clk	此信号为递增, 对应已收到的字节总数。
stat_rx_total_packets[n:0]	输出	clk	此信号为递增, 对应已收到的数据包总数。
stat_rx_total_good_bytes[13:0]	输出	clk	此信号为递增, 对应已收到的有效字节总数。仅当完全收到不含错误的数据包时, 该值才为非 0 值。
stat_rx_total_good_packets	输出	clk	此信号为递增, 对应已收到的有效数据包总数。仅当完全收到不含错误的数据包时, 该值才为非 0 值。

表 16: 统计数据接口 - RX 路径信号 (续)

名称	I/O	时钟域	描述
stat_rx_packet_bad_fcs	输出	clk	此信号为递增值，对应具有 FCS 错误的数据包（所含字节数为 64 到 <code>ctl_rx_max_packet_len</code> 之间）数。
stat_rx_packet_64_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 64 字节）数。
stat_rx_packet_65_127_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 65 到 127 字节）数。
stat_rx_packet_128_255_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 128 到 255 字节）数。
stat_rx_packet_256_511_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 256 到 511 字节）数。
stat_rx_packet_512_1023_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 512 到 1,023 字节）数。
stat_rx_packet_1024_1518_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 1,024 到 1,518 字节）数。
stat_rx_packet_1519_1522_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 1,519 到 1,522 字节）数。
stat_rx_packet_1523_1548_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 1,523 到 1,548 字节）数。
stat_rx_packet_1549_2047_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 1,549 到 2,047 字节）数。
stat_rx_packet_2048_4095_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 2,048 到 4,095 字节）数。
stat_rx_packet_4096_8191_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 4,096 到 8,191 字节）数。
stat_rx_packet_8192_9215_bytes	输出	clk	此信号为递增值，对应已收到的有效和无效数据包（含 8,192 到 9,215 字节）数。
stat_rx_packet_small[n:0]	输出	clk	此信号为递增值，对应所含数据小于 64 字节的所有包数。小于 16 字节的包将被丢弃。
stat_rx_packet_large	输出	clk	此信号为递增值，对应所含数据超过 9,215 字节的所有包数。
stat_rx_unicast	输出	clk	此信号为递增值，对应有效单播数据包数。
stat_rx_multicast	输出	clk	此信号为递增值，对应有效多播数据包数。
stat_rx_broadcast	输出	clk	此信号为递增值，对应有效广播数据包数。
stat_rx_oversize	输出	clk	此信号为递增值，对应含有效 FCS 且长度超过 <code>ctl_rx_max_packet_len</code> 的数据包数。
stat_rx_toolong	输出	clk	此信号为递增值，对应含有效和无效 FCS 且长度超过 <code>ctl_rx_max_packet_len</code> 的数据包数。
stat_rx_undersize[n:0]	输出	clk	此信号为递增值，对应含有效 FCS 且长度小于 <code>stat_rx_min_packet_len</code> 的数据包数。
stat_rx_fragment[n:0]	输出	clk	此信号为递增值，对应含无效 FCS 且长度小于 <code>ctl_rx_min_packet_len</code> 的数据包数。
stat_rx_vlan	输出	clk	此信号为递增值，对应含有效 802.1Q 标签的 VLAN 数据包数。
stat_rx_inrangeerr	输出	clk	此信号为递增值，对应含“Length”（长度）字段错误但具有有效 FCS 的数据包数。
stat_rx_jabber	输出	clk	此信号为递增值，对应含无效 FCS 且长度超过 <code>ctl_rx_max_packet_len</code> 的数据包数。
stat_rx_pause	输出	clk	此信号为递增值，对应含有效 FCS 的 802.3x Ethernet MAC 暂停包数。

表 16: 统计数据接口 - RX 路径信号 (续)

名称	I/O	时钟域	描述
stat_rx_user_pause	输出	clk	此信号为递增值，对应含有效 FCS 的基于优先级的暂停包数。
stat_tx_total_bytes[6 5 3 2:0]	输出	clk	此信号为递增值，对应已发射的字节总数。
stat_tx_total_packets	输出	clk	此信号为递增值，对应已发射的数据包总数。
stat_tx_total_good_bytes[13:0]	输出	clk	此信号为递增值，对应已发射的有效字节总数。仅当完全发射不含错误的数据包时，该值才为非 0 值。
stat_tx_total_good_packets	输出	clk	此信号为递增值，对应已发射的有效数据包总数。
stat_tx_bad_fcs	输出	clk	此信号为递增值，对应具有 FCS 错误的数据包（大于 64 字节）数。
stat_tx_packet_64_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 64 字节）数。
stat_tx_packet_65_127_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 65 到 127 字节）数。
stat_tx_packet_128_255_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 128 到 255 字节）数。

表 17: 统计数据接口 - TX 路径信号

名称	I/O	时钟域	描述
stat_tx_total_bytes[6 5 3 2:0]	输出	clk	此信号为递增值，对应已发射的字节总数。
stat_tx_total_packets	输出	clk	此信号为递增值，对应已发射的数据包总数。
stat_tx_total_good_bytes[13:0]	输出	clk	此信号为递增值，对应已发射的有效字节总数。仅当完全发射不含错误的数据包时，该值才为非 0 值。
stat_tx_total_good_packets	输出	clk	此信号为递增值，对应已发射的有效数据包总数。
stat_tx_bad_fcs	输出	clk	此信号为递增值，对应具有 FCS 错误的数据包（大于 64 字节）数。
stat_tx_packet_64_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 64 字节）数。
stat_tx_packet_65_127_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 65 到 127 字节）数。
stat_tx_packet_128_255_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 128 到 255 字节）数。
stat_tx_packet_256_511_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 256 到 511 字节）数。
stat_tx_packet_512_1023_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 512 到 1,023 字节）数。
stat_tx_packet_1024_1518_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 1,024 到 1,518 字节）数。
stat_tx_packet_1519_1522_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 1,519 到 1,522 字节）数。
stat_tx_packet_1523_1548_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 1,523 到 1,548 字节）数。
stat_tx_packet_1549_2047_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 1,549 到 2,047 字节）数。
stat_tx_packet_2048_4095_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 2,048 到 4,095 字节）数。
stat_tx_packet_4096_8191_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 4,096 到 8,191 字节）数。

表 17: 统计数据接口 - TX 路径信号 (续)

名称	I/O	时钟域	描述
stat_tx_packet_8192_9215_bytes	输出	clk	此信号为递增值，对应已发射的有效和无效数据包（含 8,192 到 9,215 字节）数。
stat_tx_packet_small	输出	clk	此信号为递增值，对应所含数据小于 64 字节的所有包数。
stat_tx_packet_large	输出	clk	此信号为递增值，对应所含数据超过 9,215 字节的所有包数。
stat_tx_unicast	输出	clk	此信号为递增值，对应有效单播数据包数。
stat_tx_multicast	输出	clk	此信号为递增值，对应有效多播数据包数。
stat_tx_broadcast	输出	clk	此信号为递增值，对应有效广播数据包数。
stat_tx_vlan	输出	clk	此信号为递增值，对应含有效 802.1Q 标签的 VLAN 数据包数。
stat_tx_pause	输出	clk	此信号为递增值，对应含有效 FCS 的 802.3x Ethernet MAC 暂停包数。
stat_tx_user_pause	输出	clk	此信号为递增值，对应含有效 FCS 的基于优先级的暂停包数。
stat_tx_frame_error	输出	clk	此信号为递增值，对应已设置 tx_errin（用于指示 EOP 中止）的数据包数。

暂停接口端口

暂停接口 - 控制端口

下表描述了控制信号。

表 18: 暂停接口 - 控制端口

名称	I/O	时钟域	描述
ctl_rx_pause_enable[8:0]	输入	rx_serdes_clk	RX 暂停使能信号。此输入用于启用对应优先级的暂停量的处理。 注释： 此信号仅影响 RX 用户接口，不影响暂停处理逻辑。
ctl_tx_pause_enable[8:0]	输入	clk	TX 暂停使能信号。此输入用于启用对应优先级的暂停量的处理。此信号用于对暂停包的发射进行门控。

暂停接口 - RX 路径端口

下表描述了 RX 路径端口。

表 19: 暂停接口 - RX 路径端口

名称	I/O	时钟域	描述
ctl_rx_enable_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制包处理。
ctl_rx_check_mcast_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制多播目标地址处理。
ctl_rx_check_ucast_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制单播目标地址处理。
ctl_rx_pause_da_ucast[47:0]	输入	rx_serdes_clk	暂停处理的单播目标地址。
ctl_rx_check_sa_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制源地址处理。
ctl_rx_pause_sa[47:0]	输入	rx_serdes_clk	暂停处理的源地址。

表 19: 暂停接口 - RX 路径端口 (续)

名称	I/O	时钟域	描述
ctl_rx_check_etype_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制以太类型 (Ethernet) 处理。
ctl_rx_check_opcode_gcp	输入	rx_serdes_clk	值为 1 即表示启用全局控制操作代码 (opcode) 处理。
ctl_rx_opcode_min_gcp[15:0]	输入	rx_serdes_clk	最小全局控制操作代码值。
ctl_rx_opcode_max_gcp[15:0]	输入	rx_serdes_clk	最大全局控制操作代码值。
ctl_rx_etype_gcp[15:0]	输入	rx_serdes_clk	全局控制处理的以太类型 (Ethernet) 字段。
ctl_rx_enable_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制包处理。
ctl_rx_check_mcast_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制多播目标地址处理。
ctl_rx_check_ucast_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制单播目标地址处理。
ctl_rx_pause_da_mcast[47:0]	输入	rx_serdes_clk	暂停处理的多播目标地址。
ctl_rx_check_sa_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制源地址处理。
ctl_rx_check_etype_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制以太类型 (Ethernet) 处理。
ctl_rx_etype_pcp[15:0]	输入	rx_serdes_clk	优先控制处理的以太类型 (Ethernet) 字段。
ctl_rx_check_opcode_pcp	输入	rx_serdes_clk	值为 1 即表示启用优先控制操作代码 (opcode) 处理。
ctl_rx_opcode_min_pcp[15:0]	输入	rx_serdes_clk	最小优先级控制操作代码值。
ctl_rx_opcode_max_pcp[15:0]	输入	rx_serdes_clk	最大优先级控制操作代码值。
ctl_rx_enable_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停包处理。
ctl_rx_check_mcast_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停多播目标地址处理。
ctl_rx_check_ucast_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停单播目标地址处理。
ctl_rx_check_sa_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停源地址处理。
ctl_rx_check_etype_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停以太类型 (Ethernet) 处理。
ctl_rx_etype_gpp[15:0]	输入	rx_serdes_clk	全局暂停处理的以太类型 (Ethernet) 字段。
ctl_rx_check_opcode_gpp	输入	rx_serdes_clk	值为 1 即表示启用全局暂停操作代码 (opcode) 处理。
ctl_rx_opcode_gpp[15:0]	输入	rx_serdes_clk	全局暂停操作代码值。
ctl_rx_enable_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停包处理。
ctl_rx_check_mcast_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停多播目标地址处理。
ctl_rx_check_ucast_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停单播目标地址处理。
ctl_rx_check_sa_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停源地址处理。
ctl_rx_check_etype_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停以太类型 (Ethernet) 处理。
ctl_rx_etype_ppp[15:0]	输入	rx_serdes_clk	优先暂停处理的以太类型 (Ethernet) 字段。
ctl_rx_check_opcode_ppp	输入	rx_serdes_clk	值为 1 即表示启用优先暂停操作代码 (opcode) 处理。
ctl_rx_opcode_ppp[15:0]	输入	rx_serdes_clk	优先暂停操作代码值。
stat_rx_pause_req[8:0]	输出	rx_serdes_clk	暂停请求信号。当 RX 收到有效的暂停帧时，它会将该总线的对应位设置为 1，其值保持为 1 不变，直至完成暂停包的处理为止。请参阅 附录 C: 暂停处理接口 以获取有关暂停接口的详细信息。
ctl_rx_pause_ack[8:0]	输入	rx_serdes_clk	暂停确认信号。该总线用于确认收到来自用户逻辑的暂停帧。请参阅 附录 C: 暂停处理接口 以获取有关暂停接口的详细信息。
ctl_rx_check_ack	输入	rx_serdes_clk	等待确认。如果此输入设置为 1，那么 40G/50G High Speed Ethernet 子系统会使用 ctl_rx_pause_ack[8:0] 总线进行暂停处理。如果此输入设置为 0，则不使用 ctl_rx_pause_ack[8:0]。

表 19: 暂停接口 - RX 路径端口 (续)

名称	I/O	时钟域	描述
ctl_rx_forward_control	输入	rx_serdes_clk	值为 1 表示 40G/50G High Speed Ethernet Subsystem 将控制包转发给您。值为 0 会导致 40G/ 50G High Speed Ethernet Subsystem 丢弃控制包。请参阅 附录 C: 暂停处理接口 以了解控制/暂停包处理。
stat_rx_pause_valid[8:0]	输出	rx_serdes_clk	该总线用于指示已收到暂停包，并且 stat_rx_pause_quanta[8:0][15:0] 总线上的关联暂停量有效，且必须将其用于暂停处理。如果收到的是 802.3x Ethernet MAC 暂停包，则 bit[8] 设置为 1。
stat_rx_pause_quanta[8:0][15:0]	输出	rx_serdes_clk	这 9 个总线用于指示在基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包，那么此暂停量将置于值 [8] 内。

暂停接口 - TX 路径端口

下表描述了 TX 路径信号。

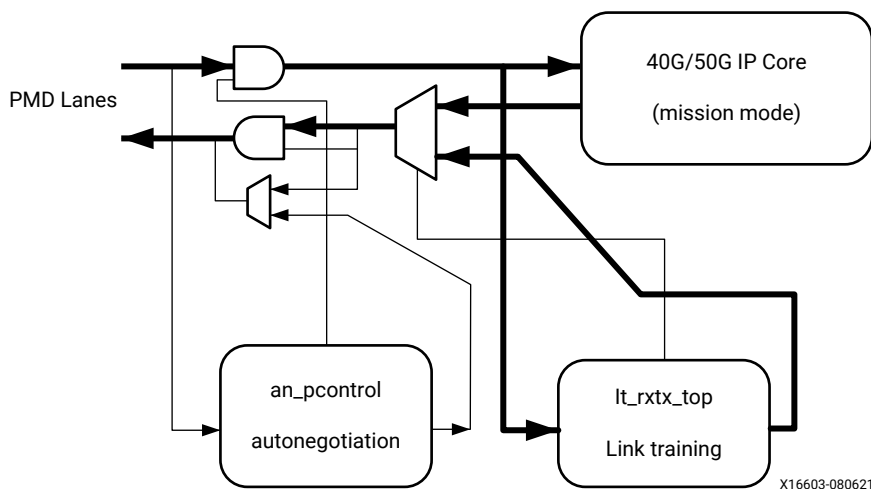
表 20: 暂停接口 - TX 路径端口

名称	I/O	时钟域	描述
ctl_tx_pause_req[8:0]	输入	clk	如果该总线的某个位设置为 1，那么 40G/50G High Speed Ethernet Subsystem 会在 ctl_tx_pause_quanta[8:0][15:0] 总线上使用关联的暂停量值来发射暂停包。如果 bit[8] 设置为 1，那么将发射全局暂停包。所有其它位设置为 1 都会导致发射优先暂停包。
ctl_tx_pause_quanta[8:0][15:0]	输入	clk	这 9 个总线用于指示在基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。
ctl_tx_pause_refresh_timer[8:0][15:0]	输入	clk	这 9 个总线用于设置在基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级，暂停包的重新发射时间。ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。
ctl_tx_da_gpp[47:0]	输入	clk	发射全局暂停包的目标地址。
ctl_tx_sa_gpp[47:0]	输入	clk	发射全局暂停包的源地址。
ctl_tx_ethertype_gpp[15:0]	输入	clk	发射全局暂停包的以太类型 (Ethertype)。
ctl_tx_opcode_gpp[15:0]	输入	clk	发射全局暂停包的操作代码。
ctl_tx_da_ppp[47:0]	输入	clk	发射优先暂停包的目标地址。
ctl_tx_sa_ppp[47:0]	输入	clk	发射优先暂停包的源地址。
ctl_tx_ethertype_ppp[15:0]	输入	clk	发射优先暂停包的以太类型 (Ethertype)。
ctl_tx_opcode_ppp[15:0]	输入	clk	发射优先暂停包的操作代码。
ctl_tx_resend_pause	输入	clk	重新发射挂起的暂停包。当此输入采样为 1 时，所有挂起的暂停包都将尽快（即，在当前进行中的包完成后）重新发射，并且重新发射计数器将复位。此输入应脉冲至 1，并且每次保持 1 个周期不变。
stat_tx_pause_valid[8:0]	输出	clk	如果该总线的某个位设置为 1，则表示 40G/50G High Speed Ethernet Subsystem 已发射暂停包。如果 bit[8] 设置为 1，则表示已发射全局暂停包。所有其它位设置为 1 都会导致发射优先暂停包。

自动协商 (AN) 和链路训练 (LT)

40G/50G IP 核支持自动协商和链路训练。含 AN 和 LT 的 40G/50G IP 核模块框图如下图所示。

图 14：含自动协商和链路训练的 40G/50G IP 核



“Auto-Negotiation”（自动协商）功能允许以太网器件将其拥有的运行模式播发至位于背板以太网链路远端的其它器件，并检测其它器件可能正在播发的对应运行模式。此自动协商功能的目的是提供一条途径，以便在 2 个器件之间交换信息，并自动配置这些器件以充分利用其功能。它的另一个目标是支持数字信号检测以确保器件连接到链路伙伴，而不是检测因串扰而产生的信号。当自动协商完成后，将根据可用运行模式报告相应的功能。

如果链路两端都支持链路训练 (LT) 功能，则会在 AN 完成后执行 LT。通常由于数字信号遍历背板可能导致产生频率相关损失，因而需要执行链路训练。该 IP 核所包含的 LT 块的主要功能是在背板链路的基础上提供寄存器信息和训练顺序，随后，SerDes 所包含的接收端电路将对这些信息进行分析。

LT 的另一项功能是将训练反馈从接收器传递到对应发射器，以便根据需要对其预加重电路（作为 SerDes 的一部分）进行调整。在此 IP 核中不包含决策制定算法。

当 AN 和 LT 完成后，数据路径将切换至任务模式（如上图所示）。

端口列表 - 自动协商

以下额外信号均用于自动协商功能。可在 `*wrapper.v` 层级中找到这些信号。

表 21：自动协商端口

端口名称	I/O	时钟域	描述和注释
an_clk	输入		自动协商电路的输入时钟。在相应版本的 README 自述文件中标示了所需的频率。此信号应为自由运行的时钟。
an_reset	输入	an_clk	同步高电平有效复位，对应于 an_clk 域。
ctl_autoneg_enable	输入	an_clk	自动协商的使能信号。

表 21：自动协商端口 (续)

端口名称	I/O	时钟域	描述和注释
ctl_autoneg_bypass	输入	an_clk	输入信号，用于禁用自动协商并绕过自动协商功能。如果断言此输入有效，将关闭自动协商，但 PCS 将连接到输出以允许执行操作。
ctl_an_nonce_seed[7:0]	输入	an_clk	8 位种子，用于初始化随机数 (nonce) 字段多项式生成器。非零值。如果该值为零 (0)，则自动协商无法运行。
ctl_an_pseudo_sel	输入	an_clk	选择位 49 随机位生成器的多项式生成器。如果此输入为 1，则多项式为 x^7+x^6+1 。如果此输入为 0，则多项式为 x^7+x^3+1 。
ctl_restart_negotiation	输入	an_clk	此输入用于触发重新启动自动协商，与当前电路状态无关。
ctl_an_local_fault	输入	an_clk	此输入信号用于设置发射链路代码字的 remote_fault 位。
用于 PAUSE 功能播发的信号			
ctl_an_pause	输入	an_clk	此输入信号用于设置发射链路代码字的 PAUSE 位 (C0)。如果该核不支持暂停，则此信号可能不存在。
ctl_an_asmdir	输入	an_clk	此输入信号用于设置发射链路代码字的 ASMDIR 位 (C1)。如果该核不支持暂停，则此信号可能不存在。
功能信号输入			
ctl_an_ability_1000base_kx	输入	an_clk	这些输入用于识别发射链路代码字中播发到链路伙伴的以太网协议功能。值为 1 表示接口支持此协议并播发此信息。
ctl_an_ability_100gbase_cr10	输入	an_clk	
ctl_an_ability_100gbase_cr4	输入	an_clk	
ctl_an_ability_100gbase_kp4	输入	an_clk	
ctl_an_ability_100gbase_kr4	输入	an_clk	
ctl_an_ability_10gbase_kr	输入	an_clk	
ctl_an_ability_10gbase_kx4	输入	an_clk	
ctl_an_ability_25gbase_krcr	输入	an_clk	
ctl_an_ability_25gbase_cr1	输入	an_clk	
ctl_an_ability_25gbase_krcr_s	输入	an_clk	
ctl_an_ability_25gbase_kr1	输入	an_clk	
ctl_an_ability_40gbase_cr4	输入	an_clk	
ctl_an_ability_40gbase_kr4	输入	an_clk	
ctl_an_ability_50gbase_cr2	输入	an_clk	
ctl_an_ability_50gbase_kr2	输入	an_clk	
ctl_lt_polynomial_select	输入	an_clk	
ctl_an_ability_2_5gbase_kx	输入	an_clk	
ctl_an_ability_5gbase_kr	输入	an_clk	
ctl_an_ability_50gbase_krcr	输入	an_clk	
ctl_an_ability_200gbase_kr4cr4	输入	an_clk	
ctl_an_ability_100gbase_kr2cr2	输入	an_clk	
ctl_an_fec_10g_request	输入	an_clk	用于在发射链路代码字中设置第 74 条 FEC 请求位。此信号仅适用于以 10 Gb/s 运行的 PMD。在以任何其他状态运行的 PMD 中，忽略此信号。

表 21：自动协商端口 (续)

端口名称	I/O	时钟域	描述和注释
ctl_an_fec_ability_override	输入	an_clk	用于在发射链路代码字中设置第 74 条 FEC 功能位。如果设置此输入，则清除发射链路代码字中的 FEC 功能位。如果该 IP 核不支持第 74 条 FEC，则此信号可能不存在。
ctl_an_cl91_fec_ability	输入	an_clk	该位用于设置第 91 条 FEC 功能。
ctl_an_cl91_fec_request	输入	an_clk	该位用于请求第 91 条 FEC。
ctl_an_fec_25g_rs_request	输入	an_clk	用于在发射链路代码字中设置 RS-FEC 请求位。此信号仅适用于以 25 Gb/s 运行的 PMD。在以任何其它状态运行的 PMD 中，忽略此信号。
ctl_an_fec_25g_baser_request	输入	an_clk	用于在发射链路代码字中设置第 74 条 FEC 请求位。此信号仅适用于以 25 Gb/s 运行的 PMD。在以任何其它状态运行的 PMD 中，忽略此信号。
stat_an_rxcdrhold	输出	an_clk	用于在自动协商期间设置 GT 的 rxcdrhold_in。
stat_an_link_cntl_1000base_kx[1:0]	输出	an_clk	来自各种以太网协议的自动协商控制器的链路控制输出。设置如下： <ul style="list-style-type: none"> · 00: DISABLE; PCS 断开连接; · 01: SCAN_FOR_CARRIER; RX 已连接到 PCS; · 11: ENABLE; PCS 已连接，可执行任务模式操作。 · 10: 不使用
stat_an_link_cntl_100gbase_cr10[1:0]	输出	an_clk	
stat_an_link_cntl_100gbase_cr4[1:0]	输出	an_clk	
stat_an_link_cntl_100gbase_kp4[1:0]	输出	an_clk	
stat_an_link_cntl_100gbase_kr4[1:0]	输出	an_clk	
stat_an_link_cntl_10gbase_kr[1:0]	输出	an_clk	
stat_an_link_cntl_10gbase_kx4[1:0]	输出	an_clk	
stat_an_link_cntl_25gbase_krcr[1:0]	输出	an_clk	
stat_an_link_cntl_25gbase_cr1[1:0]	输出	an_clk	
stat_an_link_cntl_25gbase_krcr_s[1:0]	输出	an_clk	
stat_an_link_cntl_25gbase_kr1[1:0]	输出	an_clk	
stat_an_link_cntl_40gbase_cr4[1:0]	输出	an_clk	
stat_an_link_cntl_40gbase_kr4[1:0]	输出	an_clk	
stat_an_link_cntl_50gbase_cr2[1:0]	输出	an_clk	
stat_an_link_cntl_50gbase_kr2[1:0]	输出	an_clk	
stat_an_fec_enable	输出	an_clk	用于支持在链路上使用第 74 条 FEC。
stat_an_rs_fec_enable	输出	an_clk	用于支持在链路上使用第 91 条 FEC。
stat_an_tx_pause_enable	输出	an_clk	用于支持在发射路径中进行站对站（全局）暂停包生成，以控制接收路径中的数据流。
stat_an_rx_pause_enable	输出	an_clk	用于支持在接收路径中进行站对站（全局）暂停包解释，以控制来自发射器的数据流。
stat_an_autoneg_complete	输出	an_clk	指示自动协商已完成，并且已接收到来自 PCS 的 rx 链路状态。
stat_an_parallel_detection_fault	输出	an_clk	指示自动协商期间发生并行检测故障。

表 21：自动协商端口 (续)

端口名称	I/O	时钟域	描述和注释
stat_an_lp_ability_1000base_kx	输出	an_clk	这些信号用于指示从链路伙伴播发的协议。当 stat_an_lp_ability_valid 输出信号断言有效时，这些信号全部变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_100gbase_cr10	输出	an_clk	
stat_an_lp_ability_100gbase_cr4	输出	an_clk	
stat_an_lp_ability_100gbase_kp4	输出	an_clk	
stat_an_lp_ability_100gbase_kr4	输出	an_clk	
stat_an_lp_ability_10gbase_kr	输出	an_clk	
stat_an_lp_ability_10gbase_kx4	输出	an_clk	
stat_an_lp_ability_25gbase_krcr	输出	an_clk	
stat_an_lp_ability_25gbase_krcr_s	输出	an_clk	
stat_an_lp_ability_40gbase_cr4	输出	an_clk	
stat_an_lp_ability_40gbase_kr4	输出	an_clk	
stat_an_lp_ability_25gbase_cr1	输出	an_clk	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时，此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_25gbase_kr1	输出	an_clk	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时，此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_50gbase_cr2	输出	an_clk	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时，此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_50gbase_kr2	输出	an_clk	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时，此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_pause	输出	an_clk	此信号用于指示来自链路伙伴的接收链路代码字的 PAUSE 位 (C0) 的播发值。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。
stat_an_lp_asm_dir	输出	an_clk	此信号用于指示来自链路伙伴的接收链路代码字的 ASMDIR 位 (C1) 的播发值。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。
stat_an_lp_fec_10g_ability	输出	an_clk	此信号用于指示接收链路代码字中的第 74 条 FEC 功能位的播发值，此接收链路代码字位于来自链路伙伴的对应 10 Gb/s PMD 接口上。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。
stat_an_lp_fec_10g_request	输出	an_clk	此信号用于指示接收链路代码字中的第 74 条 FEC 请求位的播发值，此接收链路代码字位于来自链路伙伴的对应 10 Gb/s PMD 接口上。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。
stat_an_lp_fec_25g_rs_request	输出	an_clk	此信号用于指示接收链路代码字中的 RS-FEC 请求位的播发值，此接收链路代码字位于来自链路伙伴的对应 25 Gb/s PMD 接口上。当输出信号断言有效时，此信号即变为有效。
stat_an_lp_fec_25g_baser_request	输出	an_clk	此信号用于指示接收链路代码字的第 74 条 FEC 请求位的播发值，此接收链路代码字位于来自链路伙伴的对应 25 Gb/s PMD 接口上。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。

表 21：自动协商端口 (续)

端口名称	I/O	时钟域	描述和注释
stat_an_lp_autoneg_able	输出	an_clk	此输出信号用于指示链路伙伴能够执行自动协商。当 stat_an_lp_ability_valid 输出信号断言有效时，此信号即变为有效。
stat_an_lp_ability_valid	输出	an_clk	此信号用于指示何时所有链路伙伴播发将变为有效。
an_loc_np_data[47:0]	输入	an_clk	本地下一页代码字。这是设置 loc_np 输入的情况下所使用的 48 位代码字。在此数据字段中，位 NP、ACK 和 T 以及位元位置 15、14、12 和 11 将不会随下一页代码字一起传输。这些位是在 AN IP 中生成的。但位元位置 13 处的消息协议位 MP 将进行传输。
an_lp_np_data[47:0]	输出	an_clk	链路伙伴下一页数据信号。此 48 位代码字由 AN IP 驱动，包含来自远程链路伙伴的 48 位下一页代码字。
ctl_an_loc_np	输入	an_clk	本地下一页 (Local Next Page) 信号指示符。如果此位为 1，则 AN IP 会将 loc_np_data 输入处的下一页代码字传输到远程链路伙伴。如果此位为 0，则 AN IP 不会启动下一页协议。如果链路伙伴包含多个待发送的下一页信号，并且 loc_np 位已清零，那么 AN IP 将传输空消息页面。
stat_fec_inc_cant_correct_count[3:0]	输出	rx_serdes_clk	不可纠正错误的逻辑指示。如果遇到不可纠正的数据包，则此输出信号会循环一次。此信号将在至少 16 个时钟周期内保持高电平 (High) 状态，然后转至低电平 (Low) 状态并保持至少 16 个时钟周期。每个通道 1 个时钟。
stat_fec_inc_correct_count[3:0]	输出	rx_serdes_clk	可纠正错误的逻辑指示。如果遇到可纠正的数据包，则此输出信号会循环一次。此信号将在至少 16 个时钟周期内保持高电平 (High) 状态，然后转至低电平 (Low) 状态并保持至少 16 个时钟周期。每个通道 1 个时钟。
stat_fec_lock_error[3:0]	输出	rx_serdes_clk	帧锁定失败的逻辑指示。接收器会扫描传入的数据串流中的约 10,000,000 位，通过尝试所有可能的位对齐来实现帧同步。随后，断言此信号高电平有效，直至接收器达成帧锁定。 每个通道 1 个时钟。
stat_fec_rx_lock[3:0]	输出	rx_serdes_clk	帧锁定的逻辑指示。接收器对传入比特流达成帧锁定时，断言此信号为高电平有效。每个通道 1 个时钟。
ctl_an_lp_np_ack	输入	an_clk	链路伙伴下一页确认信号。此信号用于告知 AN IP，本地主机已读取位于 lp_np_data 输出管脚处的下一页数据 (来自远端链路伙伴)。当此信号变为高电平时，表示 AN IP 确认已接收发送到远端链路伙伴的下一页代码字，并启动下一个代码字的传输。 在此期间，AN IP 会移除 lp_np 信号，直至新的下一页信息可用为止。
stat_an_loc_np_ack	输出	an_clk	此信号可指示用于接收 loc_np_data 输入管脚上显示的本地下一页数据的本地主机。当 AN IP 在 loc_np_data 输入管脚上对下一页数据进行采样时，此信号将脉冲至高电平并保持 1 个时钟周期。当本地主机检测到此信号为高电平时，它必须将位于 loc_np_data 输入管脚处的 48 位下一页代码字替换为要发送的下一个 48 位代码字。如果本地主机没有其它下一页要发送，则它必须将 loc_np 输入清零。

表 21：自动协商端口 (续)

端口名称	I/O	时钟域	描述和注释
stat_an_lp_np	输出	an_clk	链路伙伴下一页信号。此信号用于指示在 lp_np_data 输出管脚处存在来自远端链路伙伴的有效的 48 位下一页代码字。当 lp_np_ack 输入信号被驱动至高电平时，此信号将被驱动至低电平，表示本地主机已读取下一页数据。它将保持低电平，直至在 lp_np_data 输出管脚上下一个代码字变为可用为止，届时 lp_np 输出将被重新驱动至高电平。
stat_an_lp_ability_extended_fec[1:0]	输出	an_clk	此输出用于指示 Schedule 3 中所定义 FEC 扩展功能。
stat_an_lp_extended_ability_valid	输出	an_clk	当该位为 1 时，表示检测到的扩展功能有效。
stat_an_lp_rf	输出	an_clk	此位用于指示链路伙伴远端故障。
stat_an_start_tx_disable	输出	an_clk	当 ctl_autoneg_enable 为高电平且 ctl_autoneg_bypass 为低电平时，此 stat_an_start_tx_disable 信号会在自动协商的 TX_DISABLE 阶段开始时，在 1 个时钟周期内保持高电平循环。即，当自动协商进入 TX_DISABLE 状态时，此输出将保持高电平循环（持续 1 个时钟周期）。这样即可有效标示自动协商开始。
stat_an_start_an_good_check	输出	an_clk	当 ctl_autoneg_enable 为高电平且 ctl_autoneg_bypass 为低电平时，此 stat_an_start_an_good_check 信号会在自动协商的 AN_GOOD_CHECK 阶段开始时保持高电平循环（持续 1 个时钟周期）。即，当自动协商进入 AN_GOOD_CHECK 状态时，此输出将在 1 个时钟周期内保持循环高电平。这样即可有效标示链路训练开始。但如果未启用链路训练，即，如果 ctl_lt_training_enable 输入为低电平，那么 stat_an_start_an_good_check 输出可有效标示任务模式操作开始。

端口列表 - 链路训练

以下其它信号用于链路训练功能。可在 *wrapper.v 层级中找到这些信号。

表 22：链路训练端口

端口名称	I/O	时钟域	描述和注释
ctl_lt_training_enable	输入	tx_serdes_clk	启用链路训练。禁用链路训练时，所有 PCS 通道都以任务模式进行操作。
ctl_lt_restart_training	输入	tx_serdes_clk	此信号用于触发重新启动链路训练，与当前状态无关。
ctl_lt_rx_trained[4-1:0]	输入	tx_serdes_clk	此信号断言有效即可指示接收器 FIR 滤波器系数已全部设置完成，并且训练的接收器部分已完成。
stat_lt_signal_detect[4-1:0]	输出	tx_serdes_clk	此信号用于指示相应的链路训练状态机已进入 SEND_DATA 状态，在此状态下可恢复正常 PCS 操作。
stat_lt_training[4-1:0]	输出	tx_serdes_clk	此信号用于指示何时相应的链路训练状态机正在执行链路训练。
stat_lt_training_fail[4-1:0]	输出	tx_serdes_clk	此信号在链路训练期间断言有效，前提是对应的链路训练状态机在训练周期期间检测到超时。
stat_lt_frame_lock[4-1:0]	输出	tx_serdes_clk	链路训练开始后，将针对每个 PMD 通道断言这些信号有效，前提是对应的链路训练接收器能够与链路伙伴建立帧同步。
stat_lt_preset_from_rx[4-1:0]	输出	rx_serdes_clk	此信号用于反映从控制块中的链路伙伴接收到的预置控制位的值。

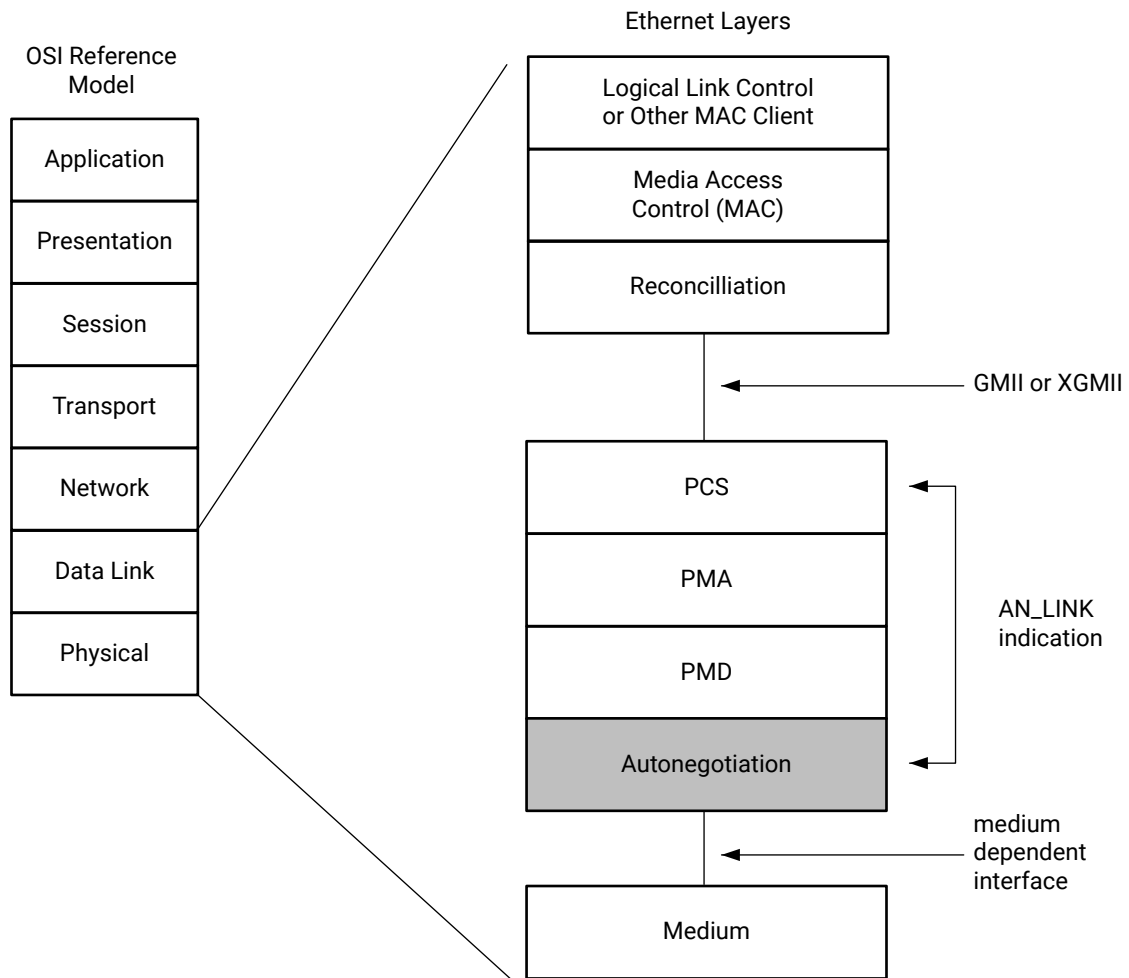
表 22：链路训练端口 (续)

端口名称	I/O	时钟域	描述和注释
stat_lt_initialize_from_rx[4-1:0]	输出	rx_serdes_clk	此信号用于反映从控制块中的链路伙伴接收到的初始化控制位的值。
stat_lt_k_p1_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k+1 系数的更新控制位。
stat_lt_k0_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k0 系数的更新控制位。
stat_lt_k_m1_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k-1 系数的更新控制位。
stat_lt_stat_p1_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k+1 系数的更新状态位。
stat_lt_stat0_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k0 系数的更新状态位。
stat_lt_stat_m1_from_rx0[1:0]	输出	rx_serdes_clk	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k-1 系数的更新状态位。
ctl_lt_pseudo_seed0[10:0]	输入	tx_serdes_clk	此 11 位信号用于植入训练模式生成器的种子。如果此种子未加载值 0，那么训练模式将不正确。
ctl_lt_preset_to_tx[4-1:0]	输入	tx_serdes_clk	此信号用于设置预置位的值，该位用于发射到训练帧的控制块中的链路伙伴。
ctl_lt_initialize_to_tx[4-1:0]	输入	tx_serdes_clk	此信号用于设置初始位的值，该位用于发射到训练帧的控制块中的链路伙伴。
ctl_lt_k_p1_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k+1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。
ctl_lt_k0_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k0 系数更新字段的值，该字段将发送到训练帧的控制块中的链路伙伴。
ctl_lt_k_m1_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k-1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。
ctl_lt_stat_p1_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k+1 系数更新状态的位，该状态用于发送到训练帧的状态块中的链路伙伴。
ctl_lt_stat0_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k0 系数更新状态的位，该状态用于发送到训练帧的状态块中的链路伙伴。
ctl_lt_stat_m1_to_tx0[1:0]	输入	tx_serdes_clk	此 2 位字段用于设置 k-1 系数更新状态的位，该状态将发送到训练帧的状态块中的链路伙伴。
stat_lt_rx_sof[4-1:0]	输出	rx_serdes_clk	此输出将在 1 个 RX SerDes 时钟周期内保持高电平，以指示链路训练帧开始。

概述

下图依据 IEEE P802.3 展示了 OSI 参考模型中自动协商功能的位置。

图 15: OSI 模型中的自动协商



X16606-080621

自动协商 IP 核满足 IEEE 802.3-2015 标准第 73 条中指定的要求，包括 IEEE P802.3by 和 25 GE 以太网联盟 Schedule 3 中指定的修订。


自动协商 IP 核的功能明确列在第 73 条中，具体来说，这些功能列在第 73.10.4 节“状态图”的“图 73-11 仲裁状态图”中。

正常任务模式操作期间，通过将链路控制输出设置为 11（二进制），SerDes 输入和输出的位工作频率通常为 10.3125 或 25.78125 Gb/s。但自动协商期间通道上所使用的双曼彻斯特编码 (Dual Manchester Encoding, DME) 比特率与任务模式操作期间所使用的比特率有很大不同。

为满足此要求，自动协商 IP 核使用过采样和过驱动来使 156.25 Mb/s 自动协商速度（DME 时钟频率 312.5 MHz）与任务模式 10.3125 或 25.78125 Gb/s 物理线速相匹配。

自动协商描述

- **autoneg_enable**: 当 `autoneg_enable` 输入信号设为 1 时, 自动协商在下列情况下自动开始执行: 在上电时、丢失载波信号时, 或者 `restart_negotiation` 输入信号从 0 到 1 循环时。所有“功能”输入信号以及 2 个输入信号 (`PAUSE` 和 `ASM_DIR`) 均绑定到低电平或高电平, 以便指示硬件功能。针对自动协商器的每个实例, `nonce_seed[7:0]` 输入都必须设置为唯一值。如果 `nonce_seed` 设为 0, 则 AN IP 将不起作用。此操作的重要意义在于, 保证在上电时不会发生死锁。如果连接在一起的 2 个链路伙伴在其 `nonce_seed[7:0]` 输入设置为相同值的情况下尝试自动协商, 那么自动协商将连续不断失败。`pseudo_sel` 输入是一种任意选择方式, 用于选择随机位生成器的多项式, 此生成器即自动协商期间所用 DME 页面的位元位置 49 中使用的随机位生成器。此输入的任何选择均有效, 并且不会导致任何有害行为。
- **链路控制**: 当自动协商开始时, 将根据各链路的对应“功能”输入的处理方法来激活各“链路控制”信号。因此, AN IP 硬件将监控对应“链路状态”信号, 以便确认可以连接的各链路的状态。如果特定链路未使用, 则对应链路控制输出处于未连接状态, 且对应链路状态输入应绑定到低电平。在此期间, AN IP 硬件会与链路伙伴建立通信链路, 并使用此链路来协商连接的功能。
- **自动协商完成**: 当自动协商完成时, 将断言 `autoneg_complete` 输出信号有效。此外, 如果要使用前向纠错硬件, 则会断言 `an_fec_enable` 输出信号有效; 如果允许发射器硬件生成 PAUSE 控制包, 则会断言 `tx_pause_en` 输出信号有效, 如果允许接收器硬件检测 PAUSE 控制包, 并且所选链路的输出链路控制设置为其任务模式值 11 (二进制), 则会断言 `rx_pause_en` 输出信号有效。

 **重要提示!** 接收到来自 PCS 的 `rx_status` 之后, 才会断言 `autoneg_complete` 信号有效。这意味着, 如果包含链路训练, 则完成链路训练并且 `rx_status` 为高电平后才会断言 `autoneg_complete` 输出信号有效。

链路训练描述

概述

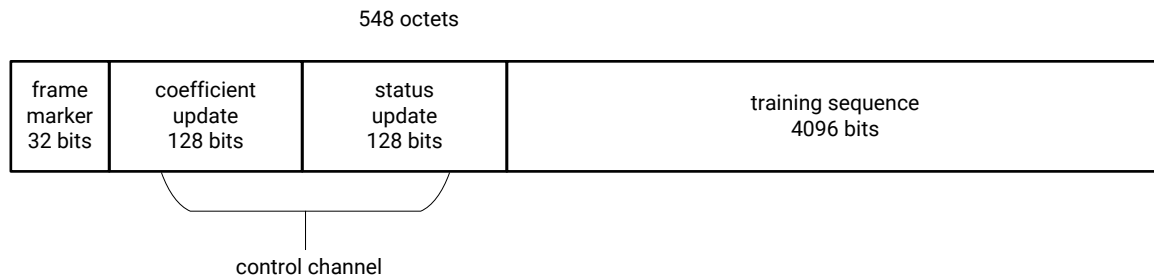
当自动协商 (AN) 集中到背板或者铜工艺后, 才会执行链路训练 (LT)。工艺选择也可以采用手动输入或并行检测方式来完成。由于频率相关损失可能导致需要进行链路培训, 而当数字信号遍历背板或铜缆时则可能会发生此类损失。LT IP 核的主要功能是在背板链路的基础上提供寄存器信息和训练顺序, 随后, 该 IP 核范围外的接收端电路将对这些信息进行分析。该 IP 核的另一项功能是将训练反馈从接收器传递到对应发射器, 以便根据需要对其均衡器电路 (不包含在此 IP 核中) 进行调整。组成该 IP 核的 2 条电路分别是链路训练接收块和链路训练发射块。

注释: 负责发射器预加重的调整工作的逻辑必须在此 IP 核外部提供。

发射

LT 发射块会构造 1 个 4,384 位帧, 其中包含帧定界符、控制通道和链路训练顺序。其格式如下所示:

图 16：链路训练帧结构



在发射状态机发射控制通道位的过程中，建议不要通过链路训练算法来更改这些位，否则接收到的位可能会有错误，从而可能导致 DME 错误。当 t_{x_SOF} 断言有效时开始发射，至少达到 288 位时间（或者约 30 ns）后停止发射。

请注意，虽然系数和状态的线速率包含 128 位时间，但这 2 个字段的实际信号处理速率将减去 8 的倍数。因此，DME 时钟速率为线速率的四分之一。

- 帧标记：帧标记包含 16 个连续的 1 后接 16 个连续的 0。在该帧剩余部分中，不会重复此模式。
- 系数和状态：由于这 2 个字段的 DME 信号处理速率减去 8 的倍数，因此每个系数和状态发射都包含 $128/8 = 16$ 位，每个位均按 15:0 的顺序编号。以下两张表按这些位的发射顺序（从位 15 开始到位 0 结束）定义这些位。

表 23：系数和更新字段位定义

位	名称	描述
15:14	Reserved	作为 0 发射，接收时忽略。
13	Preset	1 = 预置系数 0 = 正常操作
12	Initialize	1 = 初始化系数 0 = 正常操作
11:6	Reserved	作为 0 发射，接收时忽略。
5:4	Coefficient (+1) update	11 = 保留 10 = 递减 01 = 递增 00 = 保持
3:2	Coefficient (0) update	11 = 保留 10 = 递减 01 = 递增 00 = 保持
1:0	Coefficient (-1) update	11 = 保留 10 = 递减 01 = 递增 00 = 保持

表 24：状态报告字段位定义

位	名称	描述
15	Receiver ready	1 = 局部接收器已判定训练完成，并且已准备好接收数据。 0 = 局部接收器正在请求训练继续。
14:6	Reserved	作为 0 发射，接收时忽略。

表 24: 状态报告字段位定义 (续)

位	名称	描述
5:4	Coefficient (+1) update	11 = 最大 10 = 最小 01 = 已更新 00 = 未更新
3:2	Coefficient (0) update	11 = 最大 10 = 最小 01 = 已更新 00 = 未更新
1:0	Coefficient (-1) update	11 = 最大 10 = 最小 01 = 已更新 00 = 未更新

IEEE 802.3 第 72 条中定义了每个位的功能。其用途是在处理链路训练期间传递发射均衡器的调整方式。[端口描述](#)中定义了对应的信号名称。

- 训练顺序：训练顺序由伪随机位元序列 (PRBS) 组成，此序列包含 4,094 位后接 2 个 0，共计 4,096 位。PRBS 以 10.3125 或 25.78125 Gb/s 的线速率进行发射。PRBS 生成器可接收外部来源的 11 位种子。种子必须为非零。加载初始种子后，PRBS 生成器会继续运行，无需进一步干预。PRBS 生成器本身通过对应于如下多项式的电路来实现：

$$G(x) = 1 + x^9 + x^{11}$$

接收

接收块用于实现 IEEE 802.3 第 72 条中所示的帧对齐状态图。

- 帧锁定状态机：帧锁定状态机用于搜索帧标记，帧标记包含 16 个连续的 1 后接 16 个连续的 0。在 IEEE 802.3 第 72 条的图 72-4 中详细指定了该功能。达成帧锁定时，`frame_lock` 信号的值设置为 TRUE。
- 已接收的数据：接收器输出的控制通道使用在前两个表中定义的位定义以及在端口描述中定义的信号名称。如果接收特定 DME 帧期间发生 `DME_error`，则不会更新控制通道输出，但会保留最后一个接收到的有效 DME 帧的值并在接收到下一个有效 DME 帧时进行更新。

相关信息

[发射](#)
[端口描述](#)

使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试

1. 根据要播发的核配置或功能，启用功能 (Abilities) 寄存器 `CONFIGURATION_AN_ABILITY (0x00F8)`。例如，将值 `0x1E06` 写入地址 `0x00F8`。

2. 读取 CONFIGURATION_AN_CONTROL_REG1 (00E0) 寄存器。根据要求，您可启用或绕过自动协商。如已启用自动协商，则需写入随机数种子值。例如，将值 0x16D 写入地址 0x00E0。
3. 读取 CONFIGURATION_AN_CONTROL_REG2 (0x00E4) 寄存器。根据要求，您可通过写入地址 0x00E4 来启用暂停 (Pause)、FEC、RS-FEC 等。例如，将值 0x16D 写入地址 0x00E4。
4. 通过设置 CONFIGURATION_LT_CONTROL_REG1 (0x0100) 寄存器链路训练控制信号来启用链路训练选项。例如，将值 0x1 写入地址 0x0100。
5. 将 1 写入 CONFIGURATION_LT_TRAINED_REG 寄存器 (0x0104) 使 `ctl_lt_rx_trained` 转至高位。这表示接收器有限脉冲响应 (FIR) 滤波器系数已全部设置完成，并且训练的接收器部分已完成。
6. 将部分种子值写入 CONFIGURATION_LT_SEED_REG0 (0x0110) 寄存器。例如，将值 0x0605 写入地址 0x0110。
7. 将部分种子值写入 CONFIGURATION_LT_SEED_REG1 (0x0114) 寄存器。例如，将值 0x0807 写入地址 0x0114。
8. 将部分系数值（用于占位符逻辑）写入 CONFIGURATION_LT_COEFFICIENT_REG0 (0x0130) 寄存器。例如，将值 0x540 写入地址 0x0130。
9. 将部分系数值（用于占位符逻辑）写入 CONFIGURATION_LT_COEFFICIENT_REG1 (0x0134) 寄存器。例如，将值 0x555 写入地址 0x0134。
10. 向 `ctl_an_reset` 发出 `sys_reset` 或写入 1'b1，即，地址 0x0004 的第 28 位，这样自动协商块就会查找更新后的随机数种子值。
11. 保持读取 `stat_an_autoneg_complete`（地址 0x0458 的第 2 位），它可指示自动协商和链路训练是否成功完成。

前向纠错 (FEC)

40G/50G IP 核支持 25G 以太网联盟的 Schedule 3 中所定义的 3 种 FEC 运行模式：

- 无 FEC
- 第 74 条 FEC（简化循环代码 (2112, 2080)）
- 第 91 条 FEC（里所码 (528,514)）

FEC 模式将在自动协商阶段期间传递给链路伙伴。

PCS 变体

可通过相同向导生成 40G/50G Ethernet IP 子系统的仅限 PCS 变体。

功能特性

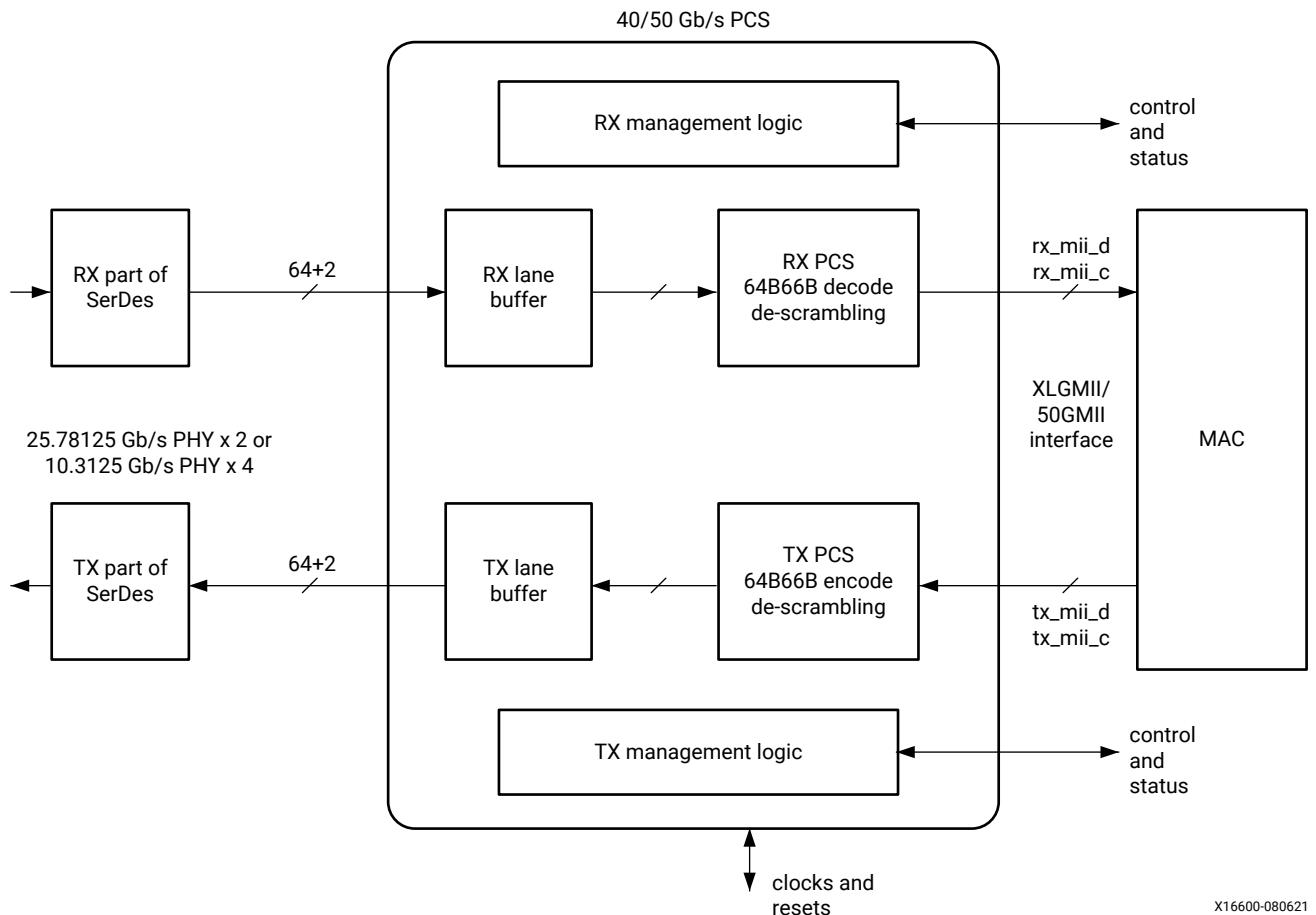
- 根据 25G 以太网联盟 Schedule 3 [<https://25gethernet.org/>] 中定义的 50 Gb/s 以太网运行要求而设计
- 根据 IEEE 802.3 第 82 条《IEEE 以太网标准》（[IEEE 802.3-2015 标准](#)）中定义的 40 Gb/s 以太网运行要求而设计
- 包含完整的 PCS 功能
- 标准 XLGMII 或 50GMII 系统侧接口
- 支持 Base-R 64b/66b 编码和解码

- 可选 Base-KR FEC
- 可选自动协商和链路训练

模块框图

下图是 40/50 Gb/s 仅限 PCS 变体的模块框图。

图 17: PCS 变体



端口列表 - 仅限 PCS

下表显示了 40G/50G PCS IP 核端口。这些是在提供仅限 PCS 选项时可用的端口。没有 FCS 功能，也不存在 AXI4-Stream 相关端口。

PCS 不包含“暂停”和“流量控制”端口。系统接口是 XLGMII/50GMII，而非 AXI4-Stream。

可在 `*wrapper.v` 层级中找到这些信号。如需了解时钟域定义，请参阅“PCS 时钟”。

表 25: PCS 变体端口

名称	I/O	时钟域	描述
收发器 I/O			
rx_serdes_data_n0	输入	rx_serdes_clk	来自通道 0 线路的串行数据；差分信号的负相
rx_serdes_data_p0	输入	rx_serdes_clk	来自通道 0 线路的串行数据；差分信号的正相
tx_serdes_data_n0	输出	tx_serdes_clk	到通道 0 线路的串行数据；差分信号的负相。
tx_serdes_data_p0	输出	tx_serdes_clk	到通道 0 线路的串行数据；差分信号的正相。
rx_serdes_data_n1	输入	rx_serdes_clk	来自通道 1 线路的串行数据；差分信号的负相
rx_serdes_data_p1	输入	rx_serdes_clk	来自通道 1 线路的串行数据；差分信号的正相
tx_serdes_data_n1	输出	tx_serdes_clk	到通道 1 线路的串行数据；差分信号的负相。
tx_serdes_data_p1	输出	tx_serdes_clk	到通道 1 线路的串行数据；差分信号的正相。
GT_reset (ctl_gt_reset_all)	输入	异步	收发器启动 FSM 的高电平有效复位。请注意，此信号还会启动整个 IP 核的复位序列。
refclk_n0	输入		SerDes 的差分参考时钟输入，负相。
refclk_p0	输入		SerDes 的差分参考时钟输入，负相。
XLGMII/50GMII 接口信号			
rx_mii_d[127:0]	输出	rx_mii_clk	接收 XLGMII/50GMII 数据总线。
rx_mii_c[15:0]	输出	rx_mii_clk	XLGMII/50GMII 接收控制总线。
rx_mii_clk	输入		接收 XLGMII/50GMII 时钟输入。
tx_mii_d[127:0]	输入	tx_mii_clk	发射 XLGMII/50GMII 数据总线。
tx_mii_c[15:0]	输入	tx_mii_clk	XLGMII/50GMII 控制总线。
tx_mii_clk	输入	tx_mii_clk	XLGMII/50GMII 发射时钟输入。
rx_mii_reset	输入		RX XLGMII/50GMII 接口的复位输入。
tx_mii_reset	输入		TX XLGMII/50GMII 接口的复位输入。
rx_serdes_clk	输入		用于对 RX PCS 核逻辑进行时钟设置的输入时钟信号。
tx_core_clk	输入		用于对 TX PCS 核逻辑进行时钟设置的输入时钟信号。
rx_reset	输入	rx_serdes_clk	与 rx_serdes_clk 逻辑关联的复位。 必须与 rx_serdes_clk 同步。
tx_reset	输入	tx_core_clk	与 tx_core_clk 逻辑关联的复位。必须与 tx_core_clk 同步。
XLGMII/50GMII 接口 - 控制/状态信号			
ctl_rx_vl_length_minus1[15:0]	输入	static	PCS 通道标记间的字数减 1（对于 RX）。根据 IEEE 802.3 中的定义，默认值应设置为 16,383。 仅当对应的复位输入断言有效时，才应更改此输入。 注释： 在 50G 核配置中启用 RS-FEC 时，此值将设置为 20479。
ctl_tx_vl_length_minus1[15:0]	输入	static	PCS 通道标记间的字数减 1（对于 TX）。根据 IEEE 802.3 中的定义，默认值应设置为 16,383。 仅当对应的复位输入断言有效时，才应更改此输入。 注释： 在 50G 核配置中启用 RS-FEC 时，此值将设置为 20479。
ctl_rx_vl_marker_id0[63:0]	输入	static	RX PCS lane0 的 PCS 通道标记。如需了解 IEEE 802.3 默认值，请参阅“RX 和 TX PCS 通道标记值”。仅当对应的复位输入断言有效时，才应更改此输入。
ctl_rx_vl_marker_id1[63:0]	输入	static	RX PCS lane1 的 PCS 通道标记。

表 25: PCS 变体端口 (续)

名称	I/O	时钟域	描述
ctl_rx_vl_marker_id2[63:0]	输入	static	RX PCS lane2 的 PCS 通道标记。
ctl_rx_vl_marker_id3[63:0]	输入	static	RX PCS lane3 的 PCS 通道标记。
ctl_tx_vl_marker_id0[63:0]	输入	static	TX PCS lane0 的 PCS 通道标记。如需了解 IEEE 802.3 默认值, 请参阅“RX 和 TX PCS 通道标记值”。仅当对应的复位输入断言有效时, 才应更改此输入。
ctl_tx_vl_marker_id1[63:0]	输入	static	TX PCS lane1 的 PCS 通道标记。
ctl_tx_vl_marker_id2[63:0]	输入	static	TX PCS lane2 的 PCS 通道标记。
ctl_tx_vl_marker_id3[63:0]	输入	static	TX PCS lane3 的 PCS 通道标记。
ctl_rx_test_pattern	输入	rx_serdes_clk	对 RX 核启用测试模式以接收扰码空闲模式。具有第三优先级。
ctl_tx_test_pattern	输入	tx_core_clk	针对 TX 核启用扰码空闲测试模式生成。值为 1 表示启用测试模式。具有第三优先级。
stat_rx_fifo_error	输出	rx_serdes_clk	接收时钟补偿 FIFO 错误指示符。值为 1 表示时钟补偿 FIFO 发生下溢或上溢。仅当恢复的时钟与局部参考时钟之间的 PPM 差超过 ± 200 ppm 时, 才会发生此状况。 如果在任意时钟周期内此输出采样为 1, 那么对应端口必须复位才能恢复正常运行。
stat_rx_local_fault	输出	rx_serdes_clk	值为 1 表示接收解码器状态机处于 RX_INIT 状态。 此输出对电平敏感。
stat_rx_hi_ber	输出	rx_serdes_clk	误码率 (BER) 过高指示符。设置为 1 时, BER 过高 (根据 IEEE 802.3 的定义)。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.32.1。 此输出对电平敏感。
stat_rx_block_lock[3:0]	输出	rx_serdes_clk	每个 PCS 通道的块锁定状态。值为 1 表示对应通道已达到第 82 条中定义的块锁定状态。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.50.7:0 和 3.51.11:0。 此输出对电平敏感。
stat_rx_error[7:0]	输出	rx_serdes_clk	测试模式不匹配数递增。任一周期内的非 0 值均表示 RX 核在测试模式下发生的不匹配次数。 仅当 ctl_rx_test_pattern 设置为 1 时, 此输出才有效。 此输出可用于根据第 82.3 条的定义生成 MDIO 寄存器 3.43.15:0。 此输出将脉冲 1 个时钟周期。
stat_rx_error_valid	输出	rx_serdes_clk	增量有效指示符。如果在任意时钟周期内此信号为 1, 则 stat_rx_error 的值有效。
stat_rx_bad_code[1:0]	输出	rx_serdes_clk	此信号为递增, 对应 64B/66B 编码违例数。此信号指示接收到的包含无效块的 64b/66b 字的数量, 或者指示是否检测到错误的 64b/66b 块序列。 此输出可用于根据第 82.3 条的定义生成 MDIO 寄存器 3.33:7:0。
stat_rx_bad_code_valid	输出	rx_serdes_clk	增量有效指示符。如果在任意时钟周期内此信号为 1, 则 stat_rx_bad_code 的值有效。
stat_rx_framing_err_0[3:0]	输出	rx_serdes_clk	针对 PCS 通道 0 检测到的同步报头错误数量的增量值。该总线的值仅在 stat_rx_framing_err_valid_0 为 1 的周期内才有效。
stat_rx_framing_err_1[3:0]	输出	rx_serdes_clk	针对 PCS 通道 1 检测到的同步报头错误数量的增量值。
stat_rx_framing_err_2[3:0]	输出	rx_serdes_clk	针对 PCS 通道 2 检测到的同步报头错误数量的增量值。
stat_rx_framing_err_3[3:0]	输出	rx_serdes_clk	针对 PCS 通道 3 检测到的同步报头错误数量的增量值。

表 25: PCS 变体端口 (续)

名称	I/O	时钟域	描述
stat_rx_valid_0	输出	rx_serdes_clk	PCS 通道 0 的增量有效指示符。如果在任意时钟周期内此信号为 1, 则 stat_rx_framing_err_0 的值有效。
stat_rx_valid_1	输出	rx_serdes_clk	PCS 通道 1 的增量有效指示符。
stat_rx_valid_2	输出	rx_serdes_clk	PCS 通道 2 的增量有效指示符。
stat_rx_valid_3	输出	rx_serdes_clk	PCS 通道 3 的增量有效指示符。
stat_rx_aligned	输出	rx_serdes_clk	所有 PCS 通道均已对齐/去歪斜。此信号表示是否所有 PCS 通道均已对齐并去歪斜。值为 1 表示所有 PCS 通道均已对齐并去歪斜。 当此信号为 1 时, 表示 RX 路径已对齐并且可接收包数据。 当此信号为 0 时, 表示存在本地故障状况。 还对应于 MDIO 寄存器位 3.50.12 (根据第 82.3 条中的定义)。此输出对电平敏感。
stat_rx_aligned_err	输出	rx_serdes_clk	通道对齐/去歪斜丢失。此信号表示 PCS 通道对齐期间发生错误或者虚拟通道对齐已丢失。值为 1 表示发生了错误。 此输出对电平敏感。
stat_rx_misaligned	输出	rx_serdes_clk	对齐错误。此信号表示通道对齐器并未在所有通道上都接收到期望的 PCS 通道标记。在所有通道上都至少接收到 1 个 PCS 通道标记后, 此信号才会断言有效。 此输出将脉冲 1 个时钟周期, 以指示错误状况。
stat_rx_status	输出	rx_serdes_clk	PCS 状态。值为 1 表示 PCS 已对齐且未处于 hi_ber 状态。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.32.12。 此输出对电平敏感。
stat_rx_vl_demuxed[3:0]	输出	rx_serdes_clk	已找到 PCS 通道标记。如果该总线的信号采样为 1, 则表示接收器已对该 PCS 通道进行了正确的逆多路复用。 此输出对电平敏感。
stat_tx_local_fault	输出	tx_core_clk	值为 1 表示发射编码器状态机当前处于 TX_INIT 状态。 此输出对电平敏感。
stat_tx_fifo_error	输出	tx_core_clk	发射时钟补偿 FIFO 错误指示符。值为 1 表示时钟补偿 FIFO 发生下溢或上溢。仅当发射器时钟与局部参考时钟之间的 PPM 差超过 ± 200 ppm 时, 才会发生此状况。 如果在任意时钟周期内此输出采样为 1, 那么对应端口必须复位才能恢复正常运行。
stat_rx_vl_number_0[1:0]	输出	rx_serdes_clk	该总线的值指示 PCS 通道 0 上出现的物理通道。 仅当 stat_rx_synced[PCS_LANES-1:0] 的对应位为 1 时, 该总线才有效。 这些输出对电平敏感。
stat_rx_vl_number_1[1:0]	输出	rx_serdes_clk	该总线的值指示 PCS 通道 1 上出现的物理通道。
stat_rx_vl_number_2[1:0]	输出	rx_serdes_clk	该总线的值指示 PCS 通道 2 上出现的物理通道。
stat_rx_vl_number_3[1:0]	输出	rx_serdes_clk	该总线的值指示 PCS 通道 3 上出现的物理通道。
stat_rx_bip_err_0	输出	rx_serdes_clk	BIP8 错误指示符 (对应 PCS 通道 0)。非 0 值表示 BIP8 签名出错。非 0 值将脉冲 1 个时钟周期。 此输出将脉冲 1 个时钟周期, 以指示错误状况。
stat_rx_bip_err_1	输出	rx_serdes_clk	BIP8 错误指示符 (对应 PCS 通道 2)。
stat_rx_bip_err_2	输出	rx_serdes_clk	BIP8 错误指示符 (对应 PCS 通道 2)。
stat_rx_bip_err_3	输出	rx_serdes_clk	BIP8 错误指示符 (对应 PCS 通道 3)。

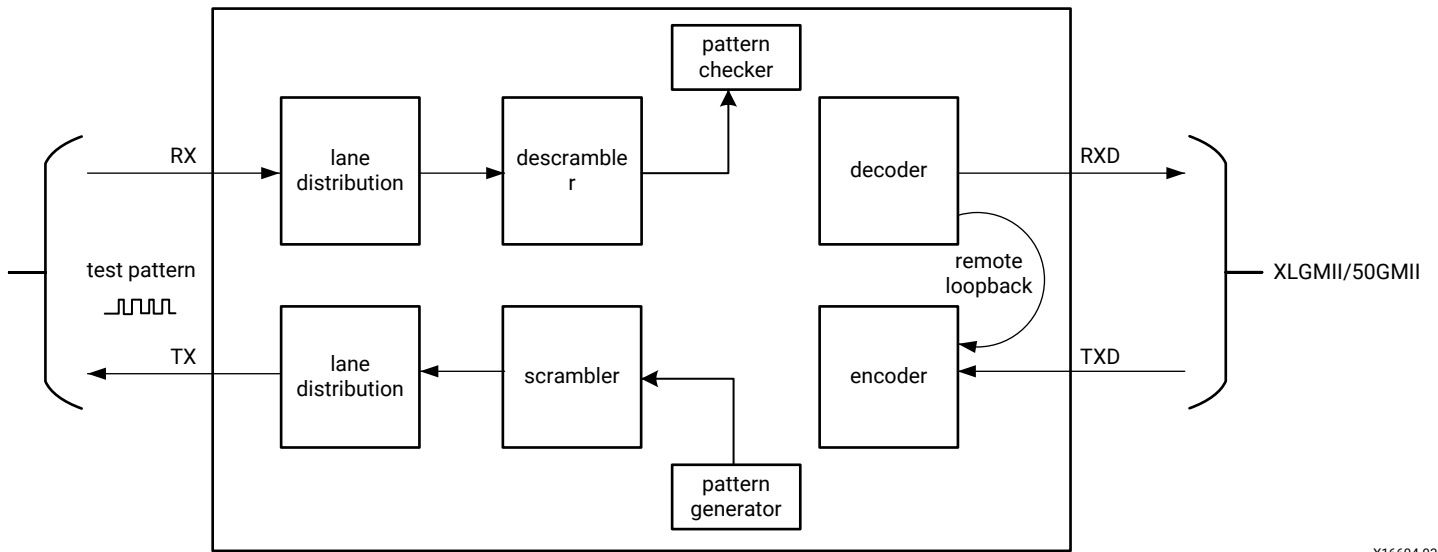
表 25: PCS 变体端口 (续)

名称	I/O	时钟域	描述
stat_rx_synced[3:0]	输出	rx_serdes_clk	字边界已同步。这些信号用于指示 PCS 通道是否已完成字边界同步。值为 1 表示对应 PCS 通道已实现字边界同步，并且已接收到 PCS 通道标记。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.52.7:0 和 3.53.11:0。 此输出对电平敏感。
stat_rx_synced_err[3:0]	输出	rx_serdes_clk	字边界同步错误。这些信号用于指示在相应的 PCS 通道内进行字边界同步期间，是否发生错误。值为 1 表示由于同步报头成帧位错误或者从未收到 PCS 通道标记，对应 PCS 通道已丢失字边界同步。 此输出对电平敏感。
stat_rx_mf_len_err[3:0]	输出	rx_serdes_clk	虚拟通道标记长度错误。这些信号用于指示相应通道内是否发生了 PCS 通道标记不匹配（即，接收到的 PCS 通道标记间隔字数并非 <code>ctl_rx_vl_length_minus1</code> ）。值为 1 表示对应通道接收 PCS 通道标记的间隔错误。 此输出将脉冲 1 个时钟周期，以指示错误状况。
stat_rx_mf_repeat_err[3:0]	输出	rx_serdes_clk	PCS 通道标记连续错误。这些信号用于指示相应通道内是否发生了 4 个连续 PCS 通道标记错误。值为 1 表示对应通道内存在错误。 此输出将脉冲 1 个时钟周期，以指示错误状况。
stat_rx_mf_err	输出	rx_serdes_clk	PCS 通道标记字错误。这些信号用于表示在相应通道内检测到构成的 PCS 通道标记字错误。值为 1 表示发生了错误。 此输出将脉冲 1 个时钟周期，以指示错误状况。
杂项状态信号和控制信号			
dclk	输入	rx_serdes_clk	动态重配置端口 (DRP) 时钟输入。通过在 Vivado® IDE 的“GT Selection and Configuration”选项卡中的“GT DRP Clock”字段中提供值，设置必需的频率。这必须是自由运行的输入时钟。
gt_loopback_in[12 6:0]	输入	异步	每个收发器的 GT 环回输入信号。请参阅 GT 用户指南。 针对 50G 单核，位宽为 6 位，针对 40G 单核或 50G 双核，位宽为 12 位。

可测试性

除 PMA 上可用的局部环回外，40G/50G PCS 还提供测试模式和远程环回功能，如下图所示。

图 18: PCS 环回



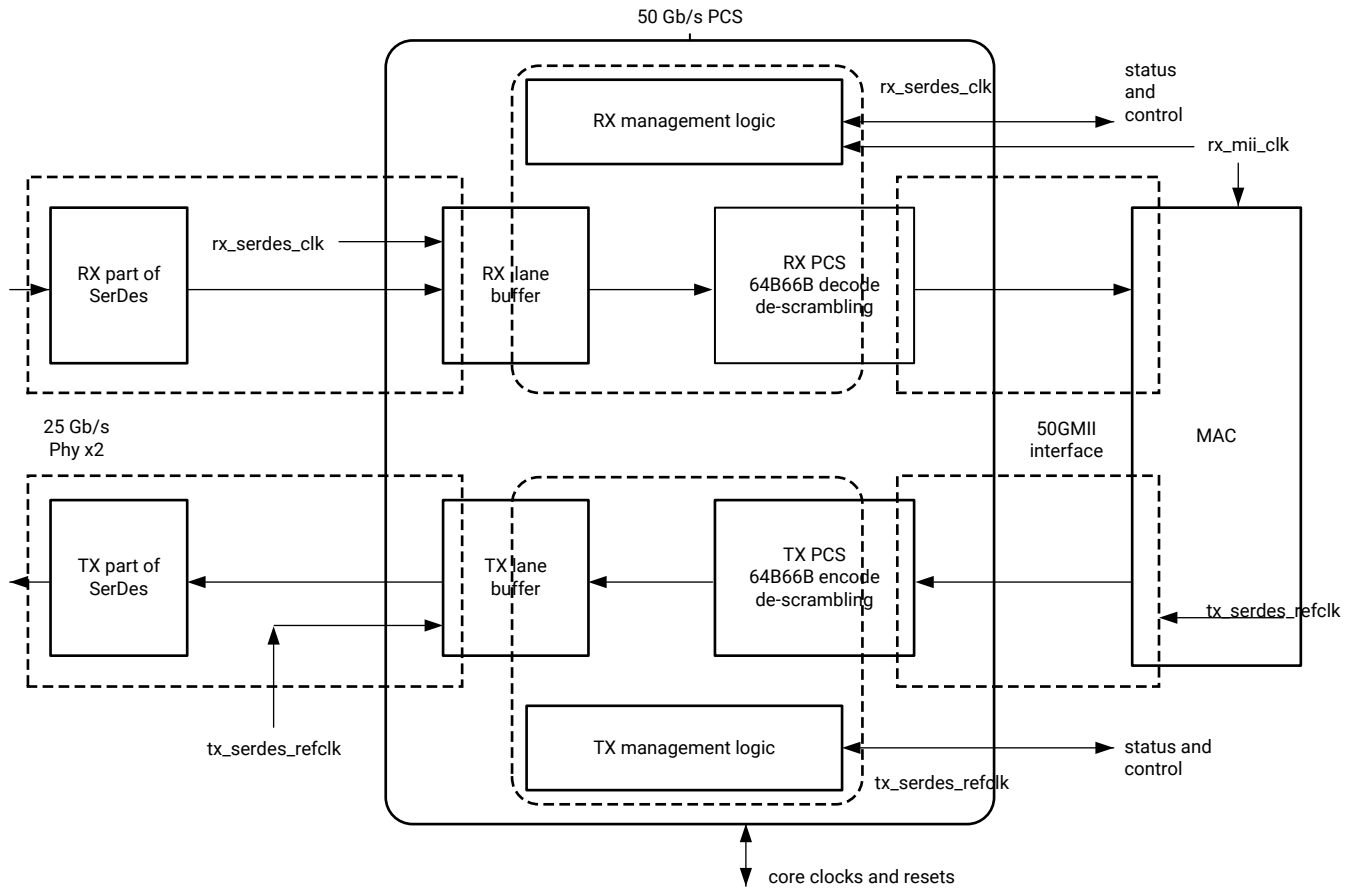
X16604-030817

启用之后，发射的测试模式为连续空闲字符，然后对这些字符进行扰码处理。

PCS 时钟

40G/50G High Speed Ethernet 子系统 PCS 使用独立的 RX 和 TX 时钟域，其中 RX 和 TX 由 3 个时钟域组成，如下图所示。

图 19: PCS 时钟



X16601-080621

XLGMII/50GMII 时钟

这些时钟用于驱动 XLGMII/50GMII 接口的逻辑。所需时钟频率由媒体独立接口 (MII) 总线宽度和数据速率来判定。例如，对于 50GMII 接口 (50 Gb/s) 和 128 位数据总线宽度，所需时钟频率为 $50e9/128 = 390.625$ MHz。对于 40G，此公式为 $40e9/128 = 312.5$ MHz。此时钟频率是根据 IEEE 以太网标准 (IEEE 802.3-2015 标准) 规范来判定的。

SerDes 时钟

- RX: 每个 SerDes 通道都有自己的已恢复时钟。此时钟用于该 SerDes 通道的所有逻辑。40G/50G High Speed Ethernet 子系统将接收到的数据从所有 SerDes 同步到 RX 核时钟域。每个 SerDes 通道都有一个时钟。SerDes 时钟频率等于数据速率除以 SerDes 宽度。例如，如果每个通道的数据速率为 25.78125 Gb/s 且 SerDes 为 66 位，则时钟频率为 $25.78125e9/66=390.625$ MHz。
- TX: TX SerDes 域与 TX 通道逻辑关联。两个 TX 收发器必须设置相同频率的时钟。按照 RX 中记录的方式计算该频率。

AXI4-Lite 寄存器空间

此 Ethernet IP 核的状态信号和控制信号可以选择通过 AXI 接口而不是宽边总线来访问。如需了解有关 AXI 寄存器中各信号的详细描述，请参阅 [端口列表 - 仅限 PCS](#)。

AXI4-Lite 端口

下图描述了 AXI 处理器接口的端口列表。

表 26: AXI4-Lite 端口

信号	I/O	描述
s_axi_aclk	输入	AXI4-Lite 时钟。范围在 10 MHz 和 300 MHz 之间
s_axi_aresetn	输入	异步低电平有效复位
s_axi_awaddr[31:0]	输入	写入地址总线
s_axi_awvalid	输入	写入地址有效
s_axi_awready	输出	写入地址确认
s_axi_wdata[31:0]	输入	写入数据总线
s_axi_wstrb[3:0]	输入	数据总线字节通道的选通信号
s_axi_wvalid	输出	写入数据有效
s_axi_wready	输出	写入数据确认
s_axi_bresp[1:0]	输出	写入传输事务响应
s_axi_bvalid	输出	写入响应有效
s_axi_bready	输入	写入响应确认
s_axi_araddr[31:0]	输入	读取地址总线
s_axi_arvalid	输入	读取地址有效
s_axi_arready	输出	读取地址确认
s_axi_rdata[31:0]	输出	读取数据输出
s_axi_rresp[1:0]	输出	读取数据响应
s_axi_rvalid	输出	读取数据/响应有效
s_axi_rready	输入	读取数据确认
pm_tick	输入	读取统计数据计数器的顶层信号；需要将 MODE_REG[30] (tick_reg_mode_sel) 设置为 0。

如需了解有关 AXI4 总线运行的其它信息，请参阅“赛灵思 AXI 存储器映射协议 1.8 版”。如前所述，可使用顶层信号 `pm_tick`（而不是配置寄存器 `TICK_REG`）来读取统计数据计数器。在此情况下，配置寄存器 `MODE_REG` 的位 30 (`tick_reg_mode_sel`) 应设置为 0。如果 `tick_reg_mode_sel` 设置为 1，则使用 `tick_reg` 来读取统计数据计数器。

基本页面

以太网寄存器映射分为 3 个部分，如下所示：

表 27: 寄存器映射

地址库	地址空间名称
0x0000	IP 配置寄存器
0x0400	状态寄存器
0x0500	统计数据计数器

所有寄存器大小均为 32 位，并按 32 位地址对齐。寄存器设计为读写完整的 32 位寄存器（忽略字节写入使能）。在以下寄存器空间映射中，地址空间内的任意空缺都应被视为保留地址 (RESERVED)，如果访问此类地址，可能导致 AXI-Ctl 接口 IP 核响应错误。

当选中 AXI 接口计数器时，使用“tick”（节拍）寄存器 (TICK_REG) 读写将来自核时钟域的统计数据捕获到 AXI 时钟域上，同时将计数器清零。发出“Tick”后，计数器会包含其更新值，并且可多次读取而不会破坏此数据。

寄存器复位信号为 `s_axi_aresetn`，此信号低电平有效。此复位会将所有寄存器强制设为默认值，如这些表中所示。

配置寄存器

配置空间可为软件提供配置各种 IP 核用例的功能。部分功能为可选（例如，自动协商、链路训练和流程功能），在此类情况下，相应的寄存器将被视为 RESERVED。

十六进制地址	名称/描述链接	注释
0x0000	GT_RESET_REG: 0000	
0x0004	RESET_REG: 0004	
0x0008	MODE_REG: 0008	
0x000C	CONFIGURATION_TX_REG1: 000C	
0x0014	CONFIGURATION_RX_REG1: 0014	
0x0018	CONFIGURATION_RX_MTU: 0018	仅限 MAC+PCS 变体
0x001C	CONFIGURATION_VL_LENGTH_REG: 001C	
0x0020	TICK_REG: 0020	
0x0024	CONFIGURATION_REVISION_REG: 0024	
0x0038	CONFIGURATION_1588_REG: 0038	仅限 MAC+PCS 变体
0x0040	CONFIGURATION_TX_FLOW_CONTROL_REG1: 0040	仅限 MAC+PCS 变体
0x0044	CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG1: 0044	仅限 MAC+PCS 变体
0x0048	CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG2: 0048	仅限 MAC+PCS 变体
0x004C	CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG3: 004C	仅限 MAC+PCS 变体
0x0050	CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG4: 0050	仅限 MAC+PCS 变体
0x0054	CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG5: 0054	仅限 MAC+PCS 变体
0x0058	CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG1: 0058	仅限 MAC+PCS 变体
0x005C	CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG2: 005C	仅限 MAC+PCS 变体
0x0060	CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG3: 0060	仅限 MAC+PCS 变体
0x0064	CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG4: 0064	仅限 MAC+PCS 变体
0x0068	CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG5: 0068	仅限 MAC+PCS 变体
0x006C	CONFIGURATION_TX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 006C	仅限 MAC+PCS 变体

十六进制地址	名称/描述链接	注释
0x0070	CONFIGURATION_TX_FLOW_CONTROL_GPP_ETYPE_OP_REG: 0070	仅限 MAC+PCS 变体
0x0074	CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_LSB: 0074	仅限 MAC+PCS 变体
0x0078	CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_MSB: 0078	仅限 MAC+PCS 变体
0x007C	CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_LSB: 007C	仅限 MAC+PCS 变体
0x0080	CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_MSB: 0080	仅限 MAC+PCS 变体
0x0084	CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_LSB: 0084	仅限 MAC+PCS 变体
0x0088	CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_MSB: 0088	仅限 MAC+PCS 变体
0x008C	CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_LSB: 008C	仅限 MAC+PCS 变体
0x0090	CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_MSB: 0090	仅限 MAC+PCS 变体
0x0094	CONFIGURATION_RX_FLOW_CONTROL_REG1: 0094	仅限 MAC+PCS 变体
0x0098	CONFIGURATION_RX_FLOW_CONTROL_REG2: 0098	仅限 MAC+PCS 变体
0x009C	CONFIGURATION_RX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 009C	仅限 MAC+PCS 变体
0x00A0	CONFIGURATION_RX_FLOW_CONTROL_GPP_ETYPE_OP_REG: 00A0	仅限 MAC+PCS 变体
0x00A4	CONFIGURATION_RX_FLOW_CONTROL_GCP_PCP_TYPE_REG: 00A4	仅限 MAC+PCS 变体
0x00A8	CONFIGURATION_RX_FLOW_CONTROL_PCP_OP_REG: 00A8	仅限 MAC+PCS 变体
0x00AC	CONFIGURATION_RX_FLOW_CONTROL_GCP_OP_REG: 00AC	仅限 MAC+PCS 变体
0x00B0	CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_LSB: 00B0	仅限 MAC+PCS 变体
0x00B4	CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_MSB: 00B4	仅限 MAC+PCS 变体
0x00B8	CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_LSB: 00B8	仅限 MAC+PCS 变体
0x00BC	CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_MSB: 00BC	仅限 MAC+PCS 变体
0x00C0	CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_LSB: 00C0	仅限 MAC+PCS 变体
0x00C4	CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_MSB: 00C4	仅限 MAC+PCS 变体
0x00D0	CONFIGURATION_RSFEC_REG: 00D0	
0x00D4	CONFIGURATION_FEC_REG: 00D4	
0x00E0	CONFIGURATION_AN_CONTROL_REG1: 00E0	
0x00E4	CONFIGURATION_AN_CONTROL_REG2: 00E4	
0x00F8	CONFIGURATION_AN_ABILITY: 00F8	
0x0100	CONFIGURATION_LT_CONTROL_REG1: 0100	
0x0104	CONFIGURATION_LT_TRAINED_REG: 0104	
0x0108	CONFIGURATION_LT_PRESET_REG: 0108	
0x010C	CONFIGURATION_LT_INIT_REG: 010C	
0x0110	CONFIGURATION_LT_SEED_REG0: 0110	
0x0114	CONFIGURATION_LT_SEED_REG1: 0114	
0x0130	CONFIGURATION_LT_COEFFICIENT_REG0: 0130	
0x0134	CONFIGURATION_LT_COEFFICIENT_REG1: 0134	
0x0138	USER_REG_0: 0138	
0x013C	SWITCH_CORE_SPEED_REG: 013C	仅适用于运行时切换
0x0154	GT_WIZ_CONTROL_REG : 0154	仅适用于 Versal ACAP

状态寄存器

状态寄存器可用于指示系统运行状况。这些寄存器为只读，并且读取操作会清除相应的寄存器。

按以下条件清除状态寄存器 >

- 应用 `s_axi_aresetn` 会同时清除 TX 和 RX 状态寄存器
- 读取特定的状态寄存器时（读取时清除）
- 应用 `rx_reset` 仅清除 RX 状态寄存器
- 应用 `tx_reset` 仅清除 TX 状态寄存器

表 28：状态寄存器

十六进制地址	名称/描述链接	注释
0x0400	STAT_TX_STATUS_REG1: 0400	
0x0404	STAT_RX_STATUS_REG1: 0404	
0x0408	STAT_STATUS_REG1: 0408	
0x040C	STAT_RX_BLOCK_LOCK_REG: 040C	
0x0410	STAT_RX_LANE_SYNC_REG: 0410	
0x0414	STAT_RX_LANE_SYNC_ERR_REG: 0414	
0x0418	STAT_RX_AM_ERR_REG: 0418	
0x041C	STAT_RX_AM_LEN_ERR_REG: 041C	
0x0420	STAT_RX_AM_REPEAT_ERR_REG: 0420	
0x0424	STAT_RX_LANE_DEMUXED: 0424	
0x0428	STAT_RX_PCS_LANE_NUM_REG1: 0428	
0x043C	STAT_RX_RSPEC_STATUS_REG: 043C	
0x0440	STAT_RX_RSPEC_LANE_FILL_REG1: 0440	
0x0448	STAT_RX_FEC_STATUS_REG: 0448	
0x044C	STAT_TX_RSPEC_STATUS_REG: 044C	
0x0450	STAT_TX_FLOW_CONTROL_REG1: 0450	仅限 MAC+PCS 变体
0x0454	STAT_RX_FLOW_CONTROL_REG1: 0454	仅限 MAC+PCS 变体
0x0458	STAT_AN_STATUS: 0458	
0x045C	STAT_AN_ABILITY: 045C	
0x0460	STAT_AN_LINK_CTL: 0460	
0x0464	STAT_LT_STATUS_REG1: 0464	
0x0468	STAT_LT_STATUS_REG2: 0468	
0x046C	STAT_LT_STATUS_REG3: 046C	
0x0470	STAT_LT_STATUS_REG4: 0470	
0x0474	STAT_LT_COEFFICIENT0_REG: 0474	
0x0478	STAT_LT_COEFFICIENT1_REG: 0478	
0x047C	STAT_CORE_SPEED_REG: 047C	
0x04A0	STAT_GT_WIZ_REG: 04A0	
0x9F0	STAT_AN_LINK_CTL2: 09F0	

统计数据计数器

统计数据计数器提供流量分类和错误计数的直方图。根据 `MODE_REG[30]`（也就是 `tick_reg_mode_sel`）的值，可通过将 `pm_tick` 设为 1，或者向 `TICK_REG` 写入值 1 来读取这些计数器。

这些计数器采用一个内部累加器。写入 `TICK_REG` 寄存器会导致将累加的计数推送到可读的 `STAT_*_MSB/LSB` 寄存器，同时将累加器清零。随后，即可读取 `STAT_*_MSB/LSB` 寄存器。这样统计数据计数器中存储的所有值都会按相同时间间隔显示快照。

注释：这些可读的 `STAT_*_MSB/LSB` 寄存器不可复位。这可能导致在复位之后且在 `TICK_REG` 写入之前存在未知的数据。

`STAT_CYCLE_COUNT_MSB/LSB` 寄存器包含 `TICK_REG` 寄存器写入间隔的 RX 核时钟周期的计数。这样即可便于提供基于时间间隔的统计数据。

计数器默认宽度为 48b。计数器饱和值为 1s。计数器的值将保留直至下一次写入 `TICK_REG` 寄存器为止。

针对计数器显示的第一个地址对是 LSB 寄存器的地址或者计数的位 31:0。计数器的 MSB 位 47:32 位于 LSB 的 + 0x4 处。

将根据以下条件来清除统计数据计数器寄存器。

- 应用 `s_axi_aresetn` 将同时清除 TX 和 RX 统计数据计数器寄存器
- 应用 PM Tick 将同时清除 TX 和 RX 统计数据计数器寄存器
- 应用 `rx_reset` 将只清除 RX 统计数据计数器寄存器
- 应用 `tx_reset` 将只清除 TX 统计数据计数器寄存器

表 29：统计数据计数器

十六进制地址	名称/描述链接	注释
0x0500	STATUS_CYCLE_COUNT_LSB: 0500	
0x0504	STATUS_CYCLE_COUNT_MSB: 0504	
0x0508	STAT_RX_BIP_ERR_0_LSB: 0508	
0x050C	STAT_RX_BIP_ERR_0_MSB: 050C	
0x0510	STAT_RX_BIP_ERR_1_LSB: 0510	
0x0514	STAT_RX_BIP_ERR_1_MSB: 0514	
0x0518	STAT_RX_BIP_ERR_2_LSB: 0518	
0x051C	STAT_RX_BIP_ERR_2_MSB: 051C	
0x0520	STAT_RX_BIP_ERR_3_LSB: 0520	
0x0524	STAT_RX_BIP_ERR_3_MSB: 0524	
0x05A8	STAT_RX_FRAMING_ERR_0_LSB: 05A8	
0x05AC	STAT_RX_FRAMING_ERR_0_MSB: 05AC	
0x05B0	STAT_RX_FRAMING_ERR_1_LSB: 05B0	
0x05B4	STAT_RX_FRAMING_ERR_1_MSB: 05B4	
0x05B8	STAT_RX_FRAMING_ERR_2_LSB: 05B8	
0x05BC	STAT_RX_FRAMING_ERR_2_MSB: 05BC	
0x05C0	STAT_RX_FRAMING_ERR_3_LSB: 05C0	
0x05C4	STAT_RX_FRAMING_ERR_3_MSB: 05C4	

表 29: 统计数据计数器 (续)

十六进制地址	名称/描述链接	注释
0x0660	STAT_RX_BAD_CODE_LSB: 0660	
0x0664	STAT_RX_BAD_CODE_MSB: 0664	
0x0668	STAT_RX_ERROR_LSB: 0668	仅限 PCS 变体
0x066C	STAT_RX_ERROR_MSB: 066C	仅限 PCS 变体
0x0670	STAT_RX_RSPEC_CORRECTED_CW_INC_LSB: 0670	
0x0674	STAT_RX_RSPEC_CORRECTED_CW_INC_MSB: 0674	
0x0678	STAT_RX_RSPEC_UNCORRECTED_CW_INC_LSB: 0678	
0x067C	STAT_RX_RSPEC_UNCORRECTED_CW_INC_MSB: 067C	
0x0680	STAT_RX_RSPEC_ERR_COUNT0_INC_LSB: 0680	
0x0684	STAT_RX_RSPEC_ERR_COUNT0_INC_MSB: 0684	
0x0688	STAT_RX_RSPEC_ERR_COUNT1_INC_LSB: 0688	
0x068C	STAT_RX_RSPEC_ERR_COUNT1_INC_MSB: 068C	
0x06A0	STAT_TX_FRAME_ERROR_LSB: 06A0	仅限 MAC+PCS 变体
0x06A4	STAT_TX_FRAME_ERROR_MSB: 06A4	仅限 MAC+PCS 变体
0x0700	STAT_TX_TOTAL_PACKETS_LSB: 0700	仅限 MAC+PCS 变体
0x0704	STAT_TX_TOTAL_PACKETS_MSB: 0704	仅限 MAC+PCS 变体
0x0708	STAT_TX_TOTAL_GOOD_PACKETS_LSB: 0708	仅限 MAC+PCS 变体
0x070C	STAT_TX_TOTAL_GOOD_PACKETS_MSB: 070C	仅限 MAC+PCS 变体
0x0710	STAT_TX_TOTAL_BYTES_LSB: 0710	仅限 MAC+PCS 变体
0x0714	STAT_TX_TOTAL_BYTES_MSB: 0714	仅限 MAC+PCS 变体
0x0718	STAT_TX_TOTAL_GOOD_BYTES_LSB: 0718	仅限 MAC+PCS 变体
0x071C	STAT_TX_TOTAL_GOOD_BYTES_MSB: 071C	仅限 MAC+PCS 变体
0x0720	STAT_TX_PACKET_64_BYTES_LSB: 0720	仅限 MAC+PCS 变体
0x0724	STAT_TX_PACKET_64_BYTES_MSB: 0724	仅限 MAC+PCS 变体
0x0728	STAT_TX_PACKET_65_127_BYTES_LSB: 0728	仅限 MAC+PCS 变体
0x072C	STAT_TX_PACKET_65_127_BYTES_MSB: 072C	仅限 MAC+PCS 变体
0x0730	STAT_TX_PACKET_128_255_BYTES_LSB: 0730	仅限 MAC+PCS 变体
0x0734	STAT_TX_PACKET_128_255_BYTES_MSB: 0734	仅限 MAC+PCS 变体
0x0738	STAT_TX_PACKET_256_511_BYTES_LSB: 0738	仅限 MAC+PCS 变体
0x073C	STAT_TX_PACKET_256_511_BYTES_MSB: 073C	仅限 MAC+PCS 变体
0x0740	STAT_TX_PACKET_512_1023_BYTES_LSB: 0740	仅限 MAC+PCS 变体
0x0744	STAT_TX_PACKET_512_1023_BYTES_MSB: 0744	仅限 MAC+PCS 变体
0x0748	STAT_TX_PACKET_1024_1518_BYTES_LSB: 0748	仅限 MAC+PCS 变体
0x074C	STAT_TX_PACKET_1024_1518_BYTES_MSB: 074C	仅限 MAC+PCS 变体
0x0750	STAT_TX_PACKET_1519_1522_BYTES_LSB: 0750	仅限 MAC+PCS 变体
0x0754	STAT_TX_PACKET_1519_1522_BYTES_MSB: 0754	仅限 MAC+PCS 变体
0x0758	STAT_TX_PACKET_1523_1548_BYTES_LSB: 0758	仅限 MAC+PCS 变体
0x075C	STAT_TX_PACKET_1523_1548_BYTES_MSB: 075C	仅限 MAC+PCS 变体
0x0760	STAT_TX_PACKET_1549_2047_BYTES_LSB: 0760	仅限 MAC+PCS 变体

表 29: 统计数据计数器 (续)

十六进制地址	名称/描述链接	注释
0x0764	STAT_TX_PACKET_1549_2047_BYTES_MSB: 0764	仅限 MAC+PCS 变体
0x0768	STAT_TX_PACKET_2048_4095_BYTES_LSB: 0768	仅限 MAC+PCS 变体
0x076C	STAT_TX_PACKET_2048_4095_BYTES_MSB: 076C	仅限 MAC+PCS 变体
0x0770	STAT_TX_PACKET_4096_8191_BYTES_LSB: 0770	仅限 MAC+PCS 变体
0x0774	STAT_TX_PACKET_4096_8191_BYTES_MSB: 0774	仅限 MAC+PCS 变体
0x0778	STAT_TX_PACKET_8192_9215_BYTES_LSB: 0778	仅限 MAC+PCS 变体
0x077C	STAT_TX_PACKET_8192_9215_BYTES_MSB: 077C	仅限 MAC+PCS 变体
0x0780	STAT_TX_PACKET_LARGE_LSB: 0780	仅限 MAC+PCS 变体
0x0784	STAT_TX_PACKET_LARGE_MSB: 0784	仅限 MAC+PCS 变体
0x0788	STAT_TX_PACKET_SMALL_LSB: 0788	仅限 MAC+PCS 变体
0x078C	STAT_TX_PACKET_SMALL_MSB: 078C	仅限 MAC+PCS 变体
0x07B8	STAT_TX_BAD_FCS_LSB: 07B8	仅限 MAC+PCS 变体
0x07BC	STAT_TX_BAD_FCS_MSB: 07BC	仅限 MAC+PCS 变体
0x07D0	STAT_TX_UNICAST_LSB: 07D0	仅限 MAC+PCS 变体
0x07D4	STAT_TX_UNICAST_MSB: 07D4	仅限 MAC+PCS 变体
0x07D8	STAT_TX_MULTICAST_LSB: 07D8	仅限 MAC+PCS 变体
0x07DC	STAT_TX_MULTICAST_MSB: 07DC	仅限 MAC+PCS 变体
0x07E0	STAT_TX_BROADCAST_LSB: 07E0	仅限 MAC+PCS 变体
0x07E4	STAT_TX_BROADCAST_MSB: 07E4	仅限 MAC+PCS 变体
0x07E8	STAT_TX_VLAN_LSB: 07E8	仅限 MAC+PCS 变体
0x07EC	STAT_TX_VLAN_MSB: 07EC	仅限 MAC+PCS 变体
0x07F0	STAT_TX_PAUSE_LSB: 07F0	仅限 MAC+PCS 变体
0x07F4	STAT_TX_PAUSE_MSB: 07F4	仅限 MAC+PCS 变体
0x07F8	STAT_TX_USER_PAUSE_LSB: 07F8	仅限 MAC+PCS 变体
0x07FC	STAT_TX_USER_PAUSE_MSB: 07FC	仅限 MAC+PCS 变体
0x0808	STAT_RX_TOTAL_PACKETS_LSB: 0808	仅限 MAC+PCS 变体
0x080C	STAT_RX_TOTAL_PACKETS_MSB: 080C	仅限 MAC+PCS 变体
0x0810	STAT_RX_TOTAL_GOOD_PACKETS_LSB: 0810	仅限 MAC+PCS 变体
0x0814	STAT_RX_TOTAL_GOOD_PACKETS_MSB: 0814	仅限 MAC+PCS 变体
0x0818	STAT_RX_TOTAL_BYTES_LSB: 0818	仅限 MAC+PCS 变体
0x081C	STAT_RX_TOTAL_BYTES_MSB: 081C	仅限 MAC+PCS 变体
0x0820	STAT_RX_TOTAL_GOOD_BYTES_LSB: 0820	仅限 MAC+PCS 变体
0x0824	STAT_RX_TOTAL_GOOD_BYTES_MSB: 0824	仅限 MAC+PCS 变体
0x0828	STAT_RX_PACKET_64_BYTES_LSB: 0828	仅限 MAC+PCS 变体
0x082C	STAT_RX_PACKET_64_BYTES_MSB: 082C	仅限 MAC+PCS 变体
0x0830	STAT_RX_PACKET_65_127_BYTES_LSB: 0830	仅限 MAC+PCS 变体
0x0834	STAT_RX_PACKET_65_127_BYTES_MSB: 0834	仅限 MAC+PCS 变体
0x0838	STAT_RX_PACKET_128_255_BYTES_LSB: 0838	仅限 MAC+PCS 变体
0x083C	STAT_RX_PACKET_128_255_BYTES_MSB: 083C	仅限 MAC+PCS 变体

表 29：统计数据计数器 (续)

十六进制地址	名称/描述链接	注释
0x0840	STAT_RX_PACKET_256_511_BYTES_LSB: 0840	仅限 MAC+PCS 变体
0x0844	STAT_RX_PACKET_256_511_BYTES_MSB: 0844	仅限 MAC+PCS 变体
0x0848	STAT_RX_PACKET_512_1023_BYTES_LSB: 0848	仅限 MAC+PCS 变体
0x084C	STAT_RX_PACKET_512_1023_BYTES_MSB: 084C	仅限 MAC+PCS 变体
0x0850	STAT_RX_PACKET_1024_1518_BYTES_LSB: 0850	仅限 MAC+PCS 变体
0x0854	STAT_RX_PACKET_1024_1518_BYTES_MSB: 0854	仅限 MAC+PCS 变体
0x0858	STAT_RX_PACKET_1519_1522_BYTES_LSB: 0858	仅限 MAC+PCS 变体
0x085C	STAT_RX_PACKET_1519_1522_BYTES_MSB: 085C	仅限 MAC+PCS 变体
0x0860	STAT_RX_PACKET_1523_1548_BYTES_LSB: 0860	仅限 MAC+PCS 变体
0x0864	STAT_RX_PACKET_1523_1548_BYTES_MSB: 0864	仅限 MAC+PCS 变体
0x0868	STAT_RX_PACKET_1549_2047_BYTES_LSB: 0868	仅限 MAC+PCS 变体
0x086C	STAT_RX_PACKET_1549_2047_BYTES_MSB: 086C	仅限 MAC+PCS 变体
0x0870	STAT_RX_PACKET_2048_4095_BYTES_LSB: 0870	仅限 MAC+PCS 变体
0x0874	STAT_RX_PACKET_2048_4095_BYTES_MSB: 0874	仅限 MAC+PCS 变体
0x0878	STAT_RX_PACKET_4096_8191_BYTES_LSB: 0878	仅限 MAC+PCS 变体
0x087C	STAT_RX_PACKET_4096_8191_BYTES_MSB: 087C	仅限 MAC+PCS 变体
0x0880	STAT_RX_PACKET_8192_9215_BYTES_LSB: 0880	仅限 MAC+PCS 变体
0x0884	STAT_RX_PACKET_8192_9215_BYTES_MSB: 0884	仅限 MAC+PCS 变体
0x0888	STAT_RX_PACKET_LARGE_LSB: 0888	仅限 MAC+PCS 变体
0x088C	STAT_RX_PACKET_LARGE_MSB: 088C	仅限 MAC+PCS 变体
0x0890	STAT_RX_PACKET_SMALL_LSB: 0890	仅限 MAC+PCS 变体
0x0894	STAT_RX_PACKET_SMALL_MSB: 0894	仅限 MAC+PCS 变体
0x0898	STAT_RX_UNDERSIZE_LSB: 0898	仅限 MAC+PCS 变体
0x089C	STAT_RX_UNDERSIZE_MSB: 089C	仅限 MAC+PCS 变体
0x08A0	STAT_RX_FRAGMENT_LSB: 08A0	仅限 MAC+PCS 变体
0x08A4	STAT_RX_FRAGMENT_MSB: 08A4	仅限 MAC+PCS 变体
0x08A8	STAT_RX_OVERSIZE_LSB: 08A8	仅限 MAC+PCS 变体
0x08AC	STAT_RX_OVERSIZE_MSB: 08AC	仅限 MAC+PCS 变体
0x08B0	STAT_RX_TOOLONG_LSB: 08B0	仅限 MAC+PCS 变体
0x08B4	STAT_RX_TOOLONG_MSB: 08B4	仅限 MAC+PCS 变体
0x08B8	STAT_RX_JABBER_LSB: 08B8	仅限 MAC+PCS 变体
0x08BC	STAT_RX_JABBER_MSB: 08BC	仅限 MAC+PCS 变体
0x08C0	STAT_RX_BAD_FCS_LSB: 08C0	仅限 MAC+PCS 变体
0x08C4	STAT_RX_BAD_FCS_MSB: 08C4	仅限 MAC+PCS 变体
0x08C8	STAT_RX_PACKET_BAD_FCS_LSB: 08C8	仅限 MAC+PCS 变体
0x08CC	STAT_RX_PACKET_BAD_FCS_MSB: 08CC	仅限 MAC+PCS 变体
0x08D0	STAT_RX_STOMPED_FCS_LSB: 08D0	仅限 MAC+PCS 变体
0x08D4	STAT_RX_STOMPED_FCS_MSB: 08D4	仅限 MAC+PCS 变体
0x08D8	STAT_RX_UNICAST_LSB: 08D8	仅限 MAC+PCS 变体

表 29: 统计数据计数器 (续)

十六进制地址	名称/描述链接	注释
0x08DC	STAT_RX_UNICAST_MSB: 08DC	仅限 MAC+PCS 变体
0x08E0	STAT_RX_MULTICAST_LSB: 08E0	仅限 MAC+PCS 变体
0x08E4	STAT_RX_MULTICAST_MSB: 08E4	仅限 MAC+PCS 变体
0x08E8	STAT_RX_BROADCAST_LSB: 08E8	仅限 MAC+PCS 变体
0x08EC	STAT_RX_BROADCAST_MSB: 08EC	仅限 MAC+PCS 变体
0x08F0	STAT_RX_VLAN_LSB: 08F0	仅限 MAC+PCS 变体
0x08F4	STAT_RX_VLAN_MSB: 08F4	仅限 MAC+PCS 变体
0x08F8	STAT_RX_PAUSE_LSB: 08F8	仅限 MAC+PCS 变体
0x08FC	STAT_RX_PAUSE_MSB: 08FC	仅限 MAC+PCS 变体
0x0900	STAT_RX_USER_PAUSE_LSB: 0900	仅限 MAC+PCS 变体
0x0904	STAT_RX_USER_PAUSE_MSB: 0904	仅限 MAC+PCS 变体
0x0908	STAT_RX_INRANGEERR_LSB: 0908	仅限 MAC+PCS 变体
0x090C	STAT_RX_INRANGEERR_MSB: 090C	仅限 MAC+PCS 变体
0x0910	STAT_RX_TRUNCATED_LSB: 0910	仅限 MAC+PCS 变体
0x0914	STAT_RX_TRUNCATED_MSB: 0914	仅限 MAC+PCS 变体
0x0918	STAT_RX_TEST_PATTERN_MISMATCH_LSB: 0918	仅限 MAC+PCS 变体
0x091C	STAT_RX_TEST_PATTERN_MISMATCH_MSB: 091C	仅限 MAC+PCS 变体
0x0920	STAT_FEC_INC_CORRECT_COUNT_LSB: 0920	
0x0924	STAT_FEC_INC_CORRECT_COUNT_MSB: 0924	
0x0928	STAT_FEC_INC_CANT_CORRECT_COUNT_LSB: 0928	
0x092C	STAT_FEC_INC_CANT_CORRECT_COUNT_MSB: 092C	

寄存器定义

配置寄存器

本节包含配置寄存器的描述。如果位元字段中所描述的功能在 IP 核中不存在，那么位元字段将还原为“RESERVED”（保留）。配置寄存器中的保留字段不接受任何写入的值，并且读取此字段时，它始终返回 0。寄存器或寄存器中的位元字段可供访问，用于执行读写 (RW)、只写 (WO) 或只读 (RO) 操作。显示的默认值均为十进制值，并在完成 `s_axi_aresetn` 后生效。

在本文档的端口列表部分中可找到每个信号的描述。

GT_RESET_REG: 0000

表 30: GT_RESET_REG: 0000

位	默认	类型	信号
0	0	RW	ctl_gt_reset_all 注释: 在写入寄存器上，这是清除信号。
1	0	RW	ctl_gt_rx_reset

表 30: GT_RESET_REG: 0000 (续)

位	默认	类型	信号
2	0	RW	ctl_gt_tx_reset

RESET_REG: 0004

在写入寄存器上，这是清除信号。在“Configuration”选项卡下选中“Include ANLT Logic”选项时，此端口可用。

表 31: RESET_REG: 0004

位	默认	类型	信号
1:0	0	RW	rx_serdes_reset
28	0	RW	ctl_an_reset
29	0	RW	tx_serdes_reset
30	0	RW	rx_reset
31	0	RW	tx_reset

MODE_REG: 0008

表 32: MODE_REG: 0008

位	默认	类型	信号
0	1	RW	en_wr_slvrr_indication
1	1	RW	en_rd_slvrr_indication
30	1	RW	tick_reg_mode_sel
31	0	RW	GT 近端 PMA 环回

注释:

1. mode_reg 寄存器的第 0 位和第 1 位支持灵活禁用或启用 slv 误差。用户可向这些位写入“0”来禁止报告 slv 错误。

CONFIGURATION_TX_REG1: 000C

表 33: CONFIGURATION_TX_REG1: 000C

位	默认	类型	信号
0	1	RW	ctl_tx_enable ¹
1	1	RW	ctl_tx_fcs_ins_enable ¹
2	0	RW	ctl_tx_ignore_fcs ¹
3	0	RW	ctl_tx_send_lfi ¹
4	0	RW	ctl_tx_send_rfi ¹
5	0	RW	ctl_tx_send_idle ¹
13:10	12	RW	ctl_tx_ipg_value ¹
14	0	RW	ctl_tx_test_pattern

表 33: CONFIGURATION_TX_REG1: 000C (续)

位	默认	类型	信号
18	0	RW	ctl_tx_custom_preamble_enable ¹

注释:

1. 仅限 MAC+PCS 变体

CONFIGURATION_RX_REG1: 0014

表 34: CONFIGURATION_RX_REG1: 0014

位	默认	类型	信号
0	1	RW	ctl_rx_enable ¹
1	1	RW	ctl_rx_delete_fcs ¹
2	0	RW	ctl_rx_ignore_fcs ¹
3	0	RW	ctl_rx_process_lfi ¹
4	1	RW	ctl_rx_check_sfd ¹
5	1	RW	ctl_rx_check_preamble ¹
6	0	RW	ctl_rx_force_resync ¹
7	0	RW	ctl_rx_test_pattern
18	0	RW	ctl_rx_custom_preamble_enable ¹

注释:

1. 仅限 MAC+PCS 变体

CONFIGURATION_RX_MTU: 0018

表 35: CONFIGURATION_RX_MTU: 0018

位	默认	类型	信号
7:0	64	RW	ctl_rx_min_packet_len
30:16	9600	RW	ctl_rx_max_packet_len

CONFIGURATION_VL_LENGTH_REG: 001C

表 36: CONFIGURATION_VL_LENGTH_REG: 001C

位	默认	类型	信号
15:0	16383	RW	ctl_tx_vl_length_minus1 注释: 在 50G 核配置中启用 RS-FEC 时, 此值将设置为 20479。
31:16	16383	RW	ctl_rx_vl_length_minus1 注释: 在 50G 核配置中启用 RS-FEC 时, 此值将设置为 20479。

TICK_REG: 0020

表 37: TICK_REG: 0020

位	默认	类型	信号
0	0	WO	tick_reg 注释：在写入寄存器上，这是清除信号。

CONFIGURATION_REVISION_REG: 0024

表 38: CONFIGURATION_REVISION_REG: 0024

位	默认	类型	信号
7:0	2	RO	major_rev
15:8	5	RO	minor_rev
31:24	0	RO	patch_rev

CONFIGURATION_1588_REG: 0038

表 39: CONFIGURATION_1588_REG: 0038

位	默认	类型	信号
0	0	RW	ctl_tx_ptp_1step_enable
1	0	RW	ctl_tx_ptp_vlane_adjust_mode
2	0	RW	ctl_ptp_transpclk_mode
26:16	0	RW	ctl_tx_ptp_latency_adjust

CONFIGURATION_TX_FLOW_CONTROL_REG1: 0040

表 40: CONFIGURATION_TX_FLOW_CONTROL_REG1: 0040

位	默认	类型	信号
8:0	0	RW	ctl_tx_pause_enable

CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG1: 0044

表 41: CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG1: 0044

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_refresh_timer0
31:16	0	RW	ctl_tx_pause_refresh_timer1

CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG2: 0048

表 42: CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG2: 0048

位	默认	类型	信号
15:0	0	RW	ctl_tx_pause_refresh_timer2
31:16	0	RW	ctl_tx_pause_refresh_timer3

CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG3: 004C

表 43: CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG3: 004C

位	默认	类型	信号
15:0	0	RW	ctl_tx_pause_refresh_timer4
31:16	0	RW	ctl_tx_pause_refresh_timer5

CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG4: 0050

表 44: CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG4: 0050

位	默认	类型	信号
15:0	0	RW	ctl_tx_pause_refresh_timer6
31:16	0	RW	ctl_tx_pause_refresh_timer7

CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG5: 0054

表 45: CONFIGURATION_TX_FLOW_CONTROL_REFRESH_REG5: 0054

位	默认	类型	信号
15:0	0	RW	ctl_tx_pause_refresh_timer8

CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG1: 0058

表 46: CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG1: 0058

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_quanta0
31:16	0	RW	ctl_tx_pause_quanta1

CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG2: 005C

表 47: CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG2: 005C

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_quanta2
31:16	0	RW	ctl_tx_pause_quanta3

CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG3: 0060

表 48: CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG3: 0060

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_quanta4
31:16	0	RW	ctl_tx_pause_quanta5

CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG4: 0064

表 49: CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG4: 0064

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_quanta6
31:16	0	RW	ctl_tx_pause_quanta7

CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG5: 0068

表 50: CONFIGURATION_TX_FLOW_CONTROL_QUANTA_REG5: 0068

位	默认	类型	信号
15:0	00	RW	ctl_tx_pause_quanta8

CONFIGURATION_TX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 006C

表 51: CONFIGURATION_TX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 006C

位	默认	类型	信号
15:0	34824	RW	ctl_tx_ethertype_ppp
31:16	257	RW	ctl_tx_opcode_ppp

CONFIGURATION_TX_FLOW_CONTROL_GPP_ETYPE_OP_REG: 0070

表 52: CONFIGURATION_TX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 0070

位	默认	类型	信号
15:0	34824	RW	ctl_tx_ethertype_gpp
31:16	1	RW	ctl_tx_opcode_gpp

CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_LSB: 0074

表 53: CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_LSB: 0074

位	默认	类型	信号
31:0	0	RW	ctl_tx_da_gpp[31:0]

CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_MSB: 0078

表 54: CONFIGURATION_TX_FLOW_CONTROL_GPP_DA_REG_MSB: 0078

位	默认	类型	信号
15:0	0	RW	ctl_tx_da_gpp[47:32]

CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_LSB: 007C

表 55: CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_LSB: 007C

位	默认	类型	信号
31:0	0	RW	ctl_tx_sa_gpp[31:0]

CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_MSB: 0080

表 56: CONFIGURATION_TX_FLOW_CONTROL_GPP_SA_REG_MSB: 0080

位	默认	类型	信号
15:0	0	RW	ctl_tx_sa_gpp[47:32]

CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_LSB: 0084

表 57: CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_LSB: 0084

位	默认	类型	信号
31:0	0	RW	ctl_tx_da_ppp[31:0]

CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_MSB: 0088

表 58: CONFIGURATION_TX_FLOW_CONTROL_PPP_DA_REG_MSB: 0088

位	默认	类型	信号
15:0	0	RW	ctl_tx_da_ppp[47:32]

CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_LSB: 008C

表 59: CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_LSB: 008C

位	默认	类型	信号
31:0	0	RW	ctl_tx_sa_ppp[31:0]

CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_MSB: 0090

表 60: CONFIGURATION_TX_FLOW_CONTROL_PPP_SA_REG_MSB: 0090

位	默认	类型	信号
15:0	0	RW	ctl_tx_da_ppp[47:32]

CONFIGURATION_RX_FLOW_CONTROL_REG1: 0094

表 61: CONFIGURATION_RX_FLOW_CONTROL_REG1: 0094

位	默认	类型	信号
8:0	0	RW	ctl_rx_pause_enable
9	0	RW	ctl_rx_forward_control
10	0	RW	ctl_rx_enable_gcp
11	0	RW	ctl_rx_enable_pcp
12	0	RW	ctl_rx_enable_gpp
13	0	RW	ctl_rx_enable_ppp
14	0	RW	ctl_rx_check_ack

CONFIGURATION_RX_FLOW_CONTROL_REG2: 0098

表 62: CONFIGURATION_RX_FLOW_CONTROL_REG2: 0098

位	默认	类型	信号
0	0	RW	ctl_rx_check_mcast_gcp
1	0	RW	ctl_rx_check_ucast_gcp
2	0	RW	ctl_rx_check_sa_gcp
3	0	RW	ctl_rx_check_etype_gcp
4	0	RW	ctl_rx_check_opcode_gcp
5	0	RW	ctl_rx_check_mcast_pcp
6	0	RW	ctl_rx_check_ucast_pcp
7	0	RW	ctl_rx_check_sa_pcp
8	0	RW	ctl_rx_check_etype_pcp
9	0	RW	ctl_rx_check_opcode_pcp
10	0	RW	ctl_rx_check_mcast_gpp
11	0	RW	ctl_rx_check_ucast_gpp
12	0	RW	ctl_rx_check_sa_gpp
13	0	RW	ctl_rx_check_etype_gpp
14	0	RW	ctl_rx_check_opcode_gpp
15	0	RW	ctl_rx_check_mcast_ppp
16	0	RW	ctl_rx_check_ucast_ppp
17	0	RW	ctl_rx_check_sa_ppp
18	0	RW	ctl_rx_check_etype_ppp

表 62: CONFIGURATION_RX_FLOW_CONTROL_REG2: 0098 (续)

位	默认	类型	信号
19	0	RW	ctl_rx_check_opcode_ppp

CONFIGURATION_RX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 009C

表 63: CONFIGURATION_RX_FLOW_CONTROL_PPP_ETYPE_OP_REG: 009C

位	默认	类型	信号
15:0	34,824	RW	ctl_rx_etype_ppp
31:16	257	RW	ctl_rx_opcode_ppp

CONFIGURATION_RX_FLOW_CONTROL_GPP_ETYPE_OP_REG: 00A0

表 64: CONFIGURATION_RX_FLOW_CONTROL_GPP_ETYPE_OP_REG: 00A0

位	默认	类型	信号
15:0	34824	RW	ctl_rx_etype_gpp
31:16	1	RW	ctl_rx_opcode_gpp

CONFIGURATION_RX_FLOW_CONTROL_GCP_PCP_TYPE_REG: 00A4

表 65: CONFIGURATION_RX_FLOW_CONTROL_GCP_PCP_TYPE_REG: 00A4

位	默认	类型	信号
15:0	34824	RW	ctl_rx_etype_gcp
31:16	34824	RW	ctl_rx_etypr_pcp

CONFIGURATION_RX_FLOW_CONTROL_PCP_OP_REG: 00A8

表 66: CONFIGURATION_RX_FLOW_CONTROL_PCP_OP_REG: 00A8

位	默认	类型	信号
15:0	257	RW	ctl_rx_opcode_min_pcp
31:16	257	RW	ctl_rx_opcode_min_pcp

CONFIGURATION_RX_FLOW_CONTROL_GCP_OP_REG: 00AC

表 67: CONFIGURATION_RX_FLOW_CONTROL_GCP_OP_REG: 00AC

位	默认	类型	信号
15:0	1	RW	ctl_rx_opcode_min_gcp
31:16	6	RW	ctl_rx_opcode_max_gcp

CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_LSB: 00B0

表 68: CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_LSB: 00A8

位	默认	类型	信号
31:0	0	RW	ctl_rx_pause_da_ucast[31:0]

CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_MSB: 00B4

表 69: CONFIGURATION_RX_FLOW_CONTROL_DA_REG1_MSB: 00B4

位	默认	类型	信号
15:0	0	RW	ctl_rx_pause_da_ucast[47:32]

CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_LSB: 00B8

表 70: CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_LSB: 00B8

位	默认	类型	信号
31:0	0	RW	ctl_rx_pause_da_mcast[31:0]

CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_MSB: 00BC

表 71: CONFIGURATION_RX_FLOW_CONTROL_DA_REG2_MSB: 00BC

位	默认	类型	信号
15:0	0	RW	ctl_rx_pause_da_mcast[47:32]

CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_LSB: 00C0

表 72: CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_LSB: 00C0

位	默认	类型	信号
31:0	0	RW	ctl_rx_pause_sa[31:0]

CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_MSB: 00C4

表 73: CONFIGURATION_RX_FLOW_CONTROL_SA_REG1_MSB: 00C4

位	默认	类型	信号
15:0	0	RW	ctl_rx_pause_sa[47:32]

CONFIGURATION_RSFEC_REG: 00D0

表 74: CONFIGURATION_RSFEC_REG: 00D0

位	默认	类型	信号
0	0	RW	ctl_rsfec_enable
2	0	RW	ctl_rx_rsfec_enable_indication
3	0	RW	ctl_rx_rsfec_enable_correction
5	0	RW	ctl_rsfec_ieee_error_indication_mode

CONFIGURATION_FEC_REG: 00D4

表 75: CONFIGURATION_FEC_REG: 00D4

位	默认	类型	信号
0	0	RW	ctl_fec_rx_enable
1	0	RW	ctl_fec_tx_enable
2	0	RW	ctl_fec_enable_error_to_pcs

CONFIGURATION_AN_CONTROL_REG1: 00E0

表 76: CONFIGURATION_AN_CONTROL_REG1: 00E0

位	默认	类型	信号
0	0	RW	ctl_autoneg_enable
1	1	RW	ctl_autoneg_bypass ¹
9:2	0	RW	ctl_an_nonce_seed
10	0	RW	ctl_an_pseudo_sel
11	0	RW	ctl_restart_negotiation
12	0	RW	ctl_an_local_fault

注释:

- 对于仿真，复位期间写入的 `ctl_autoneg_bypass` 值为 1。要采用启用 ANLT 的配置进行测试，请以 `ctl_autoneg_enable` 设为 1 且 `ctl_autoneg_bypass` 设为 0 来写入寄存器。

CONFIGURATION_AN_CONTROL_REG2: 00E4

表 77: CONFIGURATION_AN_CONTROL_REG2: 00E4

位	默认	类型	信号
0	0	RW	ctl_an_pause
1	1	RW	ctl_an_asmdir
16	0	RW	ctl_an_fec_10g_request
17	0	RW	ctl_an_fec_ability_override
18	0	RW	ctl_an_cl91_fec_request
19	0	RW	ctl_an_cl91_fec_ability

表 77: CONFIGURATION_AN_CONTROL_REG2: 00E4 (续)

位	默认	类型	信号
20	0	RW	ctl_an_fec_25g_rs_request
21	0	RW	ctl_an_fec_25g_baser_request

CONFIGURATION_AN_ABILITY: 00F8

表 78: CONFIGURATION_AN_ABILITY: 00F8

位	默认	类型	信号
0	0	RW	ctl_an_ability_1000base_kx
1	0	RW	ctl_an_ability_10gbase_kx4
2	0	RW	ctl_an_ability_10gbase_kr
3	0	RW	ctl_an_ability_40gbase_kr4
4	0	RW	ctl_an_ability_40gbase_cr4
5	0	RW	ctl_an_ability_100gbase_cr10
6	0	RW	ctl_an_ability_100gbase_kp4
7	0	RW	ctl_an_ability_100gbase_kr4
8	0	RW	ctl_an_ability_100gbase_cr4
9	0	RW	ctl_an_ability_25gbase_krcr_s
10	0	RW	ctl_an_ability_25gbase_krcr
11	0	RW	ctl_an_ability_2_5gbase_kx
12	0	RW	ctl_an_ability_5gbase_kr
13	0	RW	ctl_an_ability_50gbase_krcr
14	0	RW	ctl_an_ability_100gbase_kr2cr2
15	0	RW	ctl_an_ability_200gbase_kr4cr4
16	0	RW	ctl_an_ability_25gbase_kr1
17	0	RW	ctl_an_ability_25gbase_cr1
18	0	RW	ctl_an_ability_50gbase_kr2
19	0	RW	ctl_an_ability_50gbase_cr2

CONFIGURATION_LT_CONTROL_REG1: 0100

表 79: CONFIGURATION_LT_CONTROL_REG1: 0100

位	默认	类型	信号
0	0	RW	ctl_lt_training_enable
1	0	RW	ctl_lt_restart_training

CONFIGURATION_LT_TRAINED_REG: 0104

表 80: CONFIGURATION_LT_TRAINED_REG: 0104

位	默认	类型	信号
3:0	0	RW	ctl_lt_rx_trained

CONFIGURATION_LT_PRESET_REG: 0108

表 81: CONFIGURATION_LT_PRESET_REG: 0108

位	默认	类型	信号
3:0	0	RW	ctl_lt_preset_to_tx

CONFIGURATION_LT_INIT_REG: 010C

表 82: CONFIGURATION_LT_INIT_REG: 010C

位	默认	类型	信号
3:0	0	RW	ctl_lt_initialize_to_tx

CONFIGURATION_LT_SEED_REG0: 0110

表 83: CONFIGURATION_LT_SEED_REG0: 0110

位	默认	类型	信号
10:0	0	RW	ctl_lt_pseudo_seed0
26:16	0	RW	ctl_lt_pseudo_seed1

CONFIGURATION_LT_SEED_REG1: 0114

表 84: CONFIGURATION_LT_SEED_REG1: 0114

位	默认	类型	信号
10:0	0	RW	ctl_lt_pseudo_seed2
26:16	0	RW	ctl_lt_pseudo_seed3

CONFIGURATION_LT_COEFFICIENT_REG0: 0130

表 85: CONFIGURATION_LT_COEFFICIENT_REG0: 0130

位	默认	类型	信号
1:0	0	RW	ctl_lt_k_p1_to_tx0
3:2	0	RW	ctl_lt_k0_to_tx0
5:4	0	RW	ctl_lt_k_m1_to_tx0
7:6	0	RW	ctl_lt_stat_p1_to_tx0

表 85: CONFIGURATION_LT_COEFFICIENT_REG0: 0130 (续)

位	默认	类型	信号
9:8	0	RW	ctl_lt_stat0_to_tx0
11:10	0	RW	ctl_lt_stat_m1_to_tx0
17:16	0	RW	ctl_lt_k_p1_to_tx1
19:18	0	RW	ctl_lt_k0_to_tx1
21:20	0	RW	ctl_lt_k_m1_to_tx1
23:22	0	RW	ctl_lt_stat_p1_to_tx1
25:24	0	RW	ctl_lt_stat0_to_tx1
27:26	0	RW	ctl_lt_stat_m1_to_tx1

CONFIGURATION_LT_COEFFICIENT_REG1: 0134

表 86: CONFIGURATION_LT_COEFFICIENT_REG1: 0134

位	默认	类型	信号
1:0	0	RW	ctl_lt_k_p1_to_tx2
3:2	0	RW	ctl_lt_k0_to_tx2
5:4	0	RW	ctl_lt_k_m1_to_tx2
7:6	0	RW	ctl_lt_stat_p1_to_tx2
9:8	0	RW	ctl_lt_stat0_to_tx2
11:10	0	RW	ctl_lt_stat_m1_to_tx2
17:16	0	RW	ctl_lt_k_p1_to_tx3
19:18	0	RW	ctl_lt_k0_to_tx3
21:20	0	RW	ctl_lt_k_m1_to_tx3
23:22	0	RW	ctl_lt_stat_p1_to_tx3
25:24	0	RW	ctl_lt_stat0_to_tx3
27:26	0	RW	ctl_lt_stat_m1_to_tx3

USER_REG_0: 0138

表 87: USER_REG_0: 0138

位	默认	类型	信号
31:0	0	RW	user_reg0

SWITCH_CORE_SPEED_REG: 013C

表 88: SWITCH_CORE_SPEED_REG: 013C

位	默认	类型	信号
0	0	RW	axi_ctl_core_mode_switch

GT_WIZ_CONTROL_REG : 0154

表 89: GT_WIZ_CONTROL_REG

位	默认	类型	信号
2:0	3'b000	RW	gtwiz_loopback
7:3	5'b00000	RW	gtwiz_txprecursor
12:8	5'b00000	RW	gtwiz_txpostprecursor
19:13	7'b0000000	RW	gtwiz_txmainprecursor
20	1'b0	RW	gtwiz_rxcdrhold

注释:

- 提及的位描述（对应 gtwiz_txprecursor、gtwiz_txpostcursor、gtwiz_txmaincursor 和 gtwiz_rxcdrhold）仅对非 GTM Versal 器件有效。
- 对于 Versal GTM 器件，有效的位描述为：寄存器的位 [8:3] 对应 gtwiz_txprecursor，位 [14:9] 对应 gtwiz_txpostcursor，位 [21:15] 对应 gtwiz_txmaincursor，位 [22] 对应 gtwiz_rxcdrhold。
- 如需了解更多信息，请参阅《Versal ACAP GTY 和 GTYP 收发器架构手册》(AM002)。

状态寄存器

下表描述了此 Ethernet IP 核的状态寄存器。

某些位属于粘滞位，即一旦其值置于高电平或低电平后即加以锁存。此操作以类型 LH（已锁存至高电平）或 LL（已锁存至低电平）来指示。

STAT_TX_STATUS_REG1: 0400

表 90: STAT_TX_STATUS_REG1: 0400

位	默认	类型	信号
0	0	RO LH	stat_tx_local_fault

仅适用于“Runtime Switch”模式。写入 1 即可启用 40G 与 50G 之间的模式切换。在写入寄存器上，这是清除信号。该值用作作为执行 GT DRP 操作的传输调试模块的输入。

STAT_RX_STATUS_REG1: 0404

表 91: STAT_RX_STATUS_REG1: 0404

位	默认	类型	信号
0	1	RO LL	stat_rx_status
1	1	RO LL	stat_rx_aligned
2	0	RO LH	stat_rx_misaligned
3	0	RO LH	stat_rx_aligned_err
4	0	RO LH	stat_rx_hi_ber
5	0	RO LH	stat_rx_remote_fault ¹
6	0	RO LH	stat_rx_local_fault
7	0	RO LH	stat_rx_internal_local_fault ¹

表 91: STAT_RX_STATUS_REG1: 0404 (续)

位	默认	类型	信号
8	0	RO LH	stat_rx_received_local_fault ¹
9	0	RO LH	stat_rx_bad_preamble ¹
10	0	RO LH	stat_rx_bad_sfd ¹
11	0	RO LH	stat_rx_got_signal_os ¹

注释:

1. 仅限 MAC+PCS 变体

STAT_STATUS_REG1: 0408

表 92: STAT_STATUS_REG1: 0408

位	默认	类型	信号
0	0	RO LL	stat_rx_status ³
1	0	RO LL	stat_rx_aligned ²
2	0	RO LH	stat_rx_misaligned ²
3	0	RO LH	stat_rx_aligned_err ¹
4	0	RO LH	stat_rx_hi_ber ¹
5	0	RO LH	stat_rx_remote_fault ³
16	0	RO LH	stat_rx_local_fault

注释:

1. 仅限 MAC+PCS 变体
2. 仅限 256 位非分段式 40 Gb/s MAC+PCS 变体
3. 仅限 PCS 变体

STAT_RX_BLOCK_LOCK_REG: 040C

表 93: STAT_RX_BLOCK_LOCK_REG: 040C

位	默认	类型	信号
3:0	1	RO LL	stat_rx_block_lock

STAT_RX_LANE_SYNC_REG: 0410

表 94: STAT_RX_LANE_SYNC_REG: 0410

位	默认	类型	信号
3:0	1	RO LL	stat_rx_synced

STAT_RX_LANE_SYNC_ERR_REG: 0414

表 95: STAT_RX_LANE_SYNC_ERR: 0414

位	默认	类型	信号
3:0	0	RO LL	stat_rx_synced_err

STAT_RX_AM_ERR_REG: 0418

表 96: STAT_RX_AM_ERR_REG: 0418

位	默认	类型	信号
3:0	0	RO LL	stat_rx_mf_err

STAT_RX_AM_LEN_ERR_REG: 041C

表 97: STAT_RX_AM_LEN_ERR_REG: 041C

位	默认	类型	信号
3:0	0	RO LL	stat_rx_mf_len_err

STAT_RX_AM_REPEAT_ERR_REG: 0420

表 98: STAT_RX_AM_REPEAT_ERR_REG: 0420

位	默认	类型	信号
3:0	0	RO LH	stat_rx_mf_repeat_err

STAT_RX_LANE_DEMUXED: 0424

表 99: STAT_RX_LANE_DEMUXED: 0424

位	默认	类型	信号
3:0	0	RO	stat_rx_vl_demuxed

STAT_RX_PCS_LANE_NUM_REG1: 0428

表 100: STAT_RX_PCS_LANE_NUM_REG1: 0428

位	默认	类型	信号
1:0	0	RO	stat_rx_vl_number_0
6:5	0	RO	stat_rx_vl_number_1
11:10	0	RO	stat_rx_vl_number_2
16:15	0	RO	stat_rx_vl_number_3

STAT_RX_RSFEF_STATUS_REG: 043C

表 101: STAT_RX_RSFEF_STATUS_REG: 043C

位	默认	类型	信号
0	1	RO LH	stat_rx_rsfeF_lane_alignment_status
1	0	RO LH	stat_rx_rsfeF_lane_hi_ser
4	0	RO	stat_rx_rsfeF_am_lock0
5	0	RO	stat_rx_rsfeF_am_lock1
9:8	0	RO	stat_rx_rsfeF_lane_mapping

STAT_RX_RSFEF_LANE_FILL_REG1: 0440

表 102: STAT_RX_RSFEF_LANE_FILL_REG1: 0440

位	默认	类型	信号
13:0	0	RO	stat_rx_rsfeF_lane_fill_0
29:16	0	RO	stat_rx_rsfeF_lane_fill_1

STAT_RX_FEC_STATUS_REG: 0448

表 103: STAT_RX_FEC_STATUS_REG: 0448

位	默认	类型	信号
3:0	1	RO LL	stat_fec_rx_lock
19:16	1	RO LL	stat_fec_lock_error

STAT_TX_RSFEF_STATUS_REG: 044C

表 104: STAT_TX_RSFEF_STATUS_REG: 044C

位	默认	类型	信号
0	1	RO LL	stat_tx_rsfeF_lane_alignment_status
1	1	RO LL	stat_tx_rsfeF_block_lock

STAT_TX_FLOW_CONTROL_REG1: 0450

表 105: STAT_TX_FLOW_CONTROL_REG1: 0450

位	默认	类型	信号
8:0	0	RO LH	stat_tx_pause_valid

STAT_RX_FLOW_CONTROL_REG1: 0454

表 106: STAT_RX_FLOW_CONTROL_REG1: 0454

位	默认	类型	信号
8:0	0	RO LH	stat_rx_pause_req
17:9	0	RO LH	stat_rx_pause_valid

STAT_AN_STATUS: 0458

表 107: STAT_AN_STATUS: 0458

位	默认	类型	信号
0	0	RO	stat_an_fec_enable
1	0	RO	stat_an_rs_fec_enable
2	0	RO	stat_an_autoneg_complete
3	0	RO	stat_an_parallel_detection_fault
4	0	RO	stat_an_tx_pause_enable
5	0	RO	stat_an_rx_pause_enable
6	0	RO LH	stat_an_lp_ability_valid
7	0	RO	stat_an_lp_autoneg_able
8	0	RO	stat_an_lp_pause
9	0	RO	stat_an_lp_asm_dir
10	0	RO	stat_an_lp_rf
11	0	RO	stat_an_lp_fec_10g_ability
12	0	RO	stat_an_lp_fec_10g_request
13	0	RO LH	stat_an_lp_extended_ability_valid
17:14	0	RO	stat_an_lp_ability_extended_fec
18	0	RO	stat_an_lp_fec_25g_rs_request
19	0	RO	stat_an_lp_fec_25g_baser_request

STAT_AN_ABILITY: 045C

表 108: STAT_AN_ABILITY: 045C

位	默认	类型	信号
0	0	RO	stat_an_lp_ability_1000base_kx
1	0	RO	stat_an_lp_ability_10gbase_kx4
2	0	RO	stat_an_lp_ability_10gbase_kr
3	0	RO	stat_an_lp_ability_40gbase_kr4
4	0	RO	stat_an_lp_ability_40gbase_cr4
5	0	RO	stat_an_lp_ability_100gbase_cr10
6	0	RO	stat_an_lp_ability_100gbase_kp4
7	0	RO	stat_an_lp_ability_100gbase_kr4

表 108: STAT_AN_ABILITY: 045C (续)

位	默认	类型	信号
8	0	RO	stat_an_lp_ability_100gbase_cr4
9	0	RO	stat_an_lp_ability_25gbase_krcr_s
10	0	RO	stat_an_lp_ability_25gbase_krcr
11	0	RO	stat_an_lp_ability_2_5gbase_kx
12	0	RO	stat_an_lp_ability_5gbase_kr
13	0	RO	stat_an_lp_ability_50gbase_krcr
14	0	RO	stat_an_lp_ability_100gbase_kr2cr2
15	0	RO	stat_an_lp_ability_200gbase_kr4cr4
16	0	RO	stat_an_lp_ability_25gbase_kr1
17	0	RO	stat_an_lp_ability_25gbase_cr1
18	0	RO	stat_an_lp_ability_50gbase_kr2
19	0	RO	stat_an_lp_ability_50gbase_cr2

STAT_AN_LINK_CTL: 0460

表 109: STAT_AN_LINK_CTL: 0460

位	默认	类型	信号
1:0	0	RO	stat_an_link_cntl_1000base_kx
3:2	0	RO	stat_an_link_cntl_10gbase_kx4
5:4	0	RO	stat_an_link_cntl_10gbase_kr
7:6	0	RO	stat_an_link_cntl_40gbase_kr4
9:8	0	RO	stat_an_link_cntl_40gbase_cr4
11:10	0	RO	stat_an_link_cntl_100gbase_cr10
13:12	0	RO	stat_an_link_cntl_100gbase_kp4
15:14	0	RO	stat_an_link_cntl_100gbase_kr4
17:16	0	RO	stat_an_link_cntl_100gbase_cr4
19:18	0	RO	stat_an_link_cntl_25gbase_krcr_s
21:20	0	RO	stat_an_link_cntl_25gbase_krcr
23:22	0	RO	stat_an_link_cntl_2_5gbase_kx
25:24	0	RO	stat_an_link_cntl_5gbase_kr
27:26	0	RO	stat_an_link_cntl_50gbase_krcr
29:28	0	RO	stat_an_link_cntl_100gbase_kr2cr2
31:30	0	RO	stat_an_link_cntl_200gbase_kr4cr4

STAT_LT_STATUS_REG1: 0464

表 110: STAT_LT_STATUS_REG1: 0464

位	默认	类型	信号
3:0	0	RO	stat_lt_initialize_from_rx

表 110: STAT_LT_STATUS_REG1: 0464 (续)

位	默认	类型	信号
19:16	0	RO	stat_lt_preset_from_rx

STAT_LT_STATUS_REG2: 0468

表 111: STAT_LT_STATUS_REG2: 0468

位	默认	类型	信号
3:0	0	RO	stat_lt_training
19:16	0	RO	stat_lt_frame_lock

STAT_LT_STATUS_REG3: 046C

表 112: STAT_LT_STATUS_REG3: 046C

位	默认	类型	信号
3:0	0	RO	stat_lt_signal_detect
19:16	0	RO	stat_lt_training_fail

STAT_LT_STATUS_REG4: 0470

表 113: STAT_LT_STATUS_REG3: 0470

位	默认	类型	信号
3:0	0	RO LH	stat_lt_rx_sof

STAT_LT_COEFFICIENT0_REG: 0474

表 114: STAT_LT_COEFFICIENT0_REG: 0474

位	默认	类型	信号
1:0	0	RO	stat_lt_k_p1_from_rx0
3:2	0	RO	stat_lt_k0_from_rx0
5:4	0	RO	stat_lt_k_m1_from_rx0
7:6	0	RO	stat_lt_stat_p1_from_rx0
9:8	0	RO	stat_lt_stat0_from_rx0
11:10	0	RO	stat_lt_stat_m1_from_rx0
17:16	0	RO	stat_lt_k_p1_from_rx1
19:18	0	RO	stat_lt_k0_from_rx1
21:20	0	RO	stat_lt_k_m1_from_rx1
23:22	0	RO	stat_lt_stat_p1_from_rx1
25:24	0	RO	stat_lt_stat0_from_rx1
27:26	0	RO	stat_lt_stat_m1_from_rx1

STAT_LT_COEFFICIENT1_REG: 0478

表 115: STAT_LT_COEFFICIENT1_REG: 0478

位	默认	类型	信号
1:0	0	RO	stat_lt_k_p1_from_rx2
3:2	0	RO	stat_lt_k0_from_rx2
5:4	0	RO	stat_lt_k_m1_from_rx2
7:6	0	RO	stat_lt_stat_p1_from_rx2
9:8	0	RO	stat_lt_stat0_from_rx2
11:10	0	RO	stat_lt_stat_m1_from_rx2
17:16	0	RO	stat_lt_k_p1_from_rx3
19:18	0	RO	stat_lt_k0_from_rx3
21:20	0	RO	stat_lt_k_m1_from_rx3
23:22	0	RO	stat_lt_stat_p1_from_rx3
25:24	0	RO	stat_lt_stat0_from_rx3
27:26	0	RO	stat_lt_stat_m1_from_rx3

STAT_CORE_SPEED_REG: 047C

表 116: STAT_CORE_SPEED_REG: 047C

位	默认	类型	信号
0	GUI Configured	RO	stat_core_speed
1	GUI Configured	RO	runtime_switchable ¹

注释:

- 此寄存器仅适用于 128 位跨接式 AXI4-Stream 数据路径接口。位 1:0 定义为：
 - 00 独立 50G
 - 01 独立 40G
 - 10 运行时可切换 50G
 - 11 运行时可切换 40G

STAT_GT_WIZ_REG: 04A0

表 117: STAT_GT_WIZ_REG

位	默认	类型	信号
0	0	RO	gtwiz_reset_tx_done
1	0	RO	gtwiz_reset_rx_done

注释:

- 此寄存器表示 GT 已解复位，恢复的 TX/RX 时钟处于稳定状态。
- 用户需首先读取 0x04A0 寄存器 (STAT_GT_WIZ_REG)，并等待该值变为 2'b11。然后才能执行后续操作，如读取块锁定寄存器 (0x040C)。

STAT_AN_LINK_CTL2: 09F0

表 118: STAT_AN_LINK_CTL2: 09F0

位	默认	类型	信号
1:0	0	RO	stat_an_lp_ability_25gbase_kr1
3:2	0	RO	stat_an_link_cntl_25gbase_cr1
5:4	0	RO	stat_an_link_cntl_50gbase_kr2
7:6	0	RO	stat_an_link_cntl_50gbase_cr2

统计数据计数器

计数器为 48 位，需 2 个 32 位地址空间。下表提供了对应于 32 个 LSB 的每个计数器的第一个地址，下一个地址包含剩余 16 位 MSB（位 [15:0]）。所有计数器的默认值均为 0。

当 tick_reg（或 pm_tick，如已选中）读取计数器时，会将其清零，但寄存器仍保留其值。

每个计数器饱和值均为 FFFFFFFF（十六进制）。

STATUS_CYCLE_COUNT_LSB: 0500

表 119: STATUS_CYCLE_COUNT_LSB: 0500

位	默认	类型	信号
31:0	0	RO HIST	stat_cycle_count[31:0]

STATUS_CYCLE_COUNT_MSB: 0504

表 120: STATUS_CYCLE_COUNT_MSB: 0504

位	默认	类型	信号
15:0	0	RO HIST	stat_cycle_count[47:32]

STAT_RX_BIP_ERR_0_LSB: 0508

表 121: STAT_RX_BIP_ERR_0_LSB: 0508

位	默认	类型	信号
31:0	0	HIST	stat_rx_bip_err_0_count[31:0]

STAT_RX_BIP_ERR_0_MSB: 050C

表 122: STAT_RX_BIP_ERR_0_MSB: 050C

位	默认	类型	信号
15:0	0	HIST	stat_rx_bip_err_0_count[48-1:32]

STAT_RX_BIP_ERR_1_LSB: 0510

表 123: STAT_RX_BIP_ERR_1_LSB: 0510

位	默认	类型	信号
31:0	0	HIST	stat_rx_bip_err_1_count[31:0]

STAT_RX_BIP_ERR_1_MSB: 0514

表 124: STAT_RX_BIP_ERR_1_MSB: 0514

位	默认	类型	信号
15:0	0	HIST	stat_rx_bip_err_1_count[48-1:32]

STAT_RX_BIP_ERR_2_LSB: 0518

表 125: STAT_RX_BIP_ERR_2_LSB: 0518

位	默认	类型	信号
31:0	0	HIST	stat_rx_bip_err_2_count[31:0]

STAT_RX_BIP_ERR_2_MSB: 051C

表 126: STAT_RX_BIP_ERR_2_MSB: 051C

位	默认	类型	信号
15:0	0	HIST	stat_rx_bip_err_2_count[48-1:32]

STAT_RX_BIP_ERR_3_LSB: 0520

表 127: STAT_RX_BIP_ERR_3_LSB: 0520

位	默认	类型	信号
31:0	0	HIST	stat_rx_bip_err_3_count[31:0]

STAT_RX_BIP_ERR_3_MSB: 0524

表 128: STAT_RX_BIP_ERR_3_MSB: 0524

位	默认	类型	信号
15:0	0	HIST	stat_rx_bip_err_3_count[48-1:32]

STAT_RX_FRAMING_ERR_0_LSB: 05A8

表 129: STAT_RX_FRAMING_ERR_0_LSB: 05A8

位	默认	类型	信号
31:0	0	HIST	stat_rx_framing_err_0_count[31:0]

STAT_RX_FRAMING_ERR_0_MSB: 05AC

表 130: STAT_RX_FRAMING_ERR_0_MSB: 05AC

位	默认	类型	信号
15:0	0	HIST	stat_rx_framing_err_0_count[48-1:32]

STAT_RX_FRAMING_ERR_1_LSB: 05B0

表 131: STAT_RX_FRAMING_ERR_1_LSB: 05B0

位	默认	类型	信号
31:0	0	HIST	stat_rx_framing_err_1_count[31:0]

STAT_RX_FRAMING_ERR_1_MSB: 05B4

表 132: STAT_RX_FRAMING_ERR_1_MSB: 05B4

位	默认	类型	信号
15:0	0	HIST	stat_rx_framing_err_1_count[48-1:32]

STAT_RX_FRAMING_ERR_2_LSB: 05B8

表 133: STAT_RX_FRAMING_ERR_2_LSB: 05B8

位	默认	类型	信号
31:0	0	HIST	stat_rx_framing_err_2_count[31:0]

STAT_RX_FRAMING_ERR_2_MSB: 05BC

表 134: STAT_RX_FRAMING_ERR_2_MSB: 05BC

位	默认	类型	信号
15:0	0	HIST	stat_rx_framing_err_2_count[48-1:32]

STAT_RX_FRAMING_ERR_3_LSB: 05C0

表 135: STAT_RX_FRAMING_ERR_3_LSB: 05C0

位	默认	类型	信号
31:0	0	HIST	stat_rx_framing_err_3_count[31:0]

STAT_RX_FRAMING_ERR_3_MSB: 05C4

表 136: STAT_RX_FRAMING_ERR_3_MSB: 05C4

位	默认	类型	信号
15:0	0	HIST	stat_rx_framing_err_3_count[48-1:32]

STAT_RX_BAD_CODE_LSB: 0660

表 137: STAT_RX_BAD_CODE_LSB: 0660

位	默认	类型	信号
31:0	0	HIST	stat_rx_bad_code_count[31:0]

STAT_RX_BAD_CODE_MSB: 0664

表 138: STAT_RX_BAD_CODE_MSB: 0664

位	默认	类型	信号
15:0	0	HIST	stat_rx_bad_code_count[48-1:32]

STAT_RX_ERROR_LSB: 0668

表 139: STAT_RX_ERROR_LSB: 0668

位	默认	类型	信号
31:0	0	HIST	stat_rx_error_count[31:0]

STAT_RX_ERROR_MSB: 066C

表 140: STAT_RX_ERROR_MSB: 066C

位	默认	类型	信号
15:0	0	HIST	stat_rx_error_count[47:32]

STAT_RX_RSFECCORRECTED_CW_INC_LSB: 0670

表 141: STAT_RX_RSFECCORRECTED_CW_INC_LSB: 0670

位	默认	类型	信号
31:0	0	HIST	stat_rx_rsfec_corrected_cw_inc_count[31:0]

STAT_RX_RSFECCORRECTED_CW_INC_MSB: 0674

表 142: STAT_RX_RSFECCORRECTED_CW_INC_MSB: 0674

位	默认	类型	信号
15:0	0	HIST	stat_rx_rsfec_corrected_cw_inc_count[47:32]

STAT_RX_RSFECCORRECTED_CW_INC_LSB: 0678

表 143: STAT_RX_RSFECCORRECTED_CW_INC_LSB: 0678

位	默认	类型	信号
31:0	0	HIST	stat_rx_rsfec_uncorrected_cw_inc_count[31:0]

STAT_RX_RSFECCORRECTED_CW_INC_MSB: 067C

表 144: STAT_RX_RSFECCORRECTED_CW_INC_MSB: 067C

位	默认	类型	信号
31:0	0	HIST	stat_rx_rsfec_uncorrected_cw_inc_count[47:32]

STAT_RX_RSFECCOUNT0_INC_LSB: 0680

表 145: STAT_RX_RSFECCOUNT0_INC_LSB: 0680

位	默认	类型	信号
31:0	0	HIST	stat_rx_rsfec_err_count0_inc_count[31:0]

STAT_RX_RSFECCOUNT0_INC_MSB: 0684

表 146: STAT_RX_RSFECCOUNT0_INC_MSB: 0684

位	默认	类型	信号
15:0	0	HIST	stat_rx_rsfec_err_count0_inc_count[47:32]

STAT_RX_RSFEC_ERR_COUNT1_INC_LSB: 0688

表 147: STAT_RX_RSFEC_ERR_COUNT1_INC_LSB: 0688

位	默认	类型	信号
31:0	0	HIST	stat_rx_rsfec_err_count1_inc_count[31:0]

STAT_RX_RSFEC_ERR_COUNT1_INC_LSB: 068C

表 148: STAT_RX_RSFEC_ERR_COUNT1_INC_LSB: 068C

位	默认	类型	信号
15:0	0	HIST	stat_rx_rsfec_err_count1_inc_count[47:32]

STAT_TX_FRAME_ERROR_LSB: 06A0

表 149: STAT_TX_FRAME_ERROR_LSB: 06A0

位	默认	类型	信号
31:0	0	HIST	stat_tx_frame_error_count[31:0]

STAT_TX_FRAME_ERROR_MSB: 06A4

表 150: STAT_TX_FRAME_ERROR_MSB: 06A4

位	默认	类型	信号
15:0	0	HIST	stat_tx_frame_error_count[48-1:32]

STAT_TX_TOTAL_PACKETS_LSB: 0700

表 151: STAT_TX_TOTAL_PACKETS_LSB: 0700

位	默认	类型	信号
31:0	0	HIST	stat_tx_total_packets_count[31:0]

STAT_TX_TOTAL_PACKETS_MSB: 0704

表 152: STAT_TX_TOTAL_PACKETS_MSB: 0704

位	默认	类型	信号
15:0	0	HIST	stat_tx_total_packets_count[48-1:32]

STAT_TX_TOTAL_GOOD_PACKETS_LSB: 0708

表 153: STAT_TX_TOTAL_GOOD_PACKETS_LSB: 0708

位	默认	类型	信号
31:0	0	HIST	stat_tx_total_good_packets_count[31:0]

STAT_TX_TOTAL_GOOD_PACKETS_MSB: 070C

表 154: STAT_TX_TOTAL_GOOD_PACKETS_MSB: 070C

位	默认	类型	信号
15:0	0	HIST	stat_tx_total_good_packets_count[48-1:32]

STAT_TX_TOTAL_BYTES_LSB: 0710

表 155: STAT_TX_TOTAL_BYTES_LSB: 0710

位	默认	类型	信号
31:0	0	HIST	stat_tx_total_bytes_count[31:0]

STAT_TX_TOTAL_BYTES_MSB: 0714

表 156: STAT_TX_TOTAL_BYTES_MSB: 0714

位	默认	类型	信号
15:0	0	HIST	stat_tx_total_bytes_count[48-1:32]

STAT_TX_TOTAL_GOOD_BYTES_LSB: 0718

表 157: STAT_TX_TOTAL_GOOD_BYTES_LSB: 0718

位	默认	类型	信号
31:0	0	HIST	stat_tx_total_good_bytes_count[31:0]

STAT_TX_TOTAL_GOOD_BYTES_MSB: 071C

表 158: STAT_TX_TOTAL_GOOD_BYTES_MSB: 071C

位	默认	类型	信号
15:0	0	HIST	stat_tx_total_good_bytes_count[48-1:32]

STAT_TX_PACKET_64_BYTES_LSB: 0720

表 159: STAT_TX_PACKET_64_BYTES_LSB: 0720

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_64_bytes_count[31:0]

STAT_TX_PACKET_64_BYTES_MSB: 0724

表 160: STAT_TX_PACKET_64_BYTES_MSB: 0724

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_64_bytes_count[48-1:32]

STAT_TX_PACKET_65_127_BYTES_LSB: 0728

表 161: STAT_TX_PACKET_65_127_BYTES_LSB: 0728

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_65_127_bytes_count[31:0]

STAT_TX_PACKET_65_127_BYTES_MSB: 072C

表 162: STAT_TX_PACKET_65_127_BYTES_MSB: 072C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_65_127_bytes_count[48-1:32]

STAT_TX_PACKET_128_255_BYTES_LSB: 0730

表 163: STAT_TX_PACKET_128_255_BYTES_LSB: 0730

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_128_255_bytes_count[31:0]

STAT_TX_PACKET_128_255_BYTES_MSB: 0734

表 164: STAT_TX_PACKET_128_255_BYTES_MSB: 0734

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_128_255_bytes_count[48-1:32]

STAT_TX_PACKET_256_511_BYTES_LSB: 0738

表 165: STAT_TX_PACKET_256_511_BYTES_LSB: 0738

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_256_511_bytes_count[31:0]

STAT_TX_PACKET_256_511_BYTES_MSB: 073C

表 166: STAT_TX_PACKET_256_511_BYTES_MSB: 073C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_256_511_bytes_count[48-1:32]

STAT_TX_PACKET_512_1023_BYTES_LSB: 0740

表 167: STAT_TX_PACKET_512_1023_BYTES_LSB: 0740

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_512_1023_bytes_count[31:0]

STAT_TX_PACKET_512_1023_BYTES_MSB: 0744

表 168: STAT_TX_PACKET_512_1023_BYTES_MSB: 0744

位	默认	类型	信号
15:0		HIST	stat_tx_packet_512_1023_bytes_count[48-1:32]

STAT_TX_PACKET_1024_1518_BYTES_LSB: 0748

表 169: STAT_TX_PACKET_1024_1518_BYTES_LSB: 0748

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_1024_1518_bytes_count[31:0]

STAT_TX_PACKET_1024_1518_BYTES_MSB: 074C

表 170: STAT_TX_PACKET_1024_1518_BYTES_MSB: 074C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_1024_1518_bytes_count[48-1:32]

STAT_TX_PACKET_1519_1522_BYTES_LSB: 0750

表 171: STAT_TX_PACKET_1519_1522_BYTES_LSB: 0750

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_1519_1522_bytes_count[31:0]

STAT_TX_PACKET_1519_1522_BYTES_MSB: 0754

表 172: STAT_TX_PACKET_1519_1522_BYTES_MSB: 0754

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_1519_1522_bytes_count[48-1:32]

STAT_TX_PACKET_1523_1548_BYTES_LSB: 0758

表 173: STAT_TX_PACKET_1523_1548_BYTES_LSB: 0758

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_1523_1548_bytes_count[31:0]

STAT_TX_PACKET_1523_1548_BYTES_MSB: 075C

表 174: STAT_TX_PACKET_1523_1548_BYTES_MSB: 075C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_1523_1548_bytes_count[48-1:32]

STAT_TX_PACKET_1549_2047_BYTES_LSB: 0760

表 175: STAT_TX_PACKET_1549_2047_BYTES_LSB: 0760

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_1549_2047_bytes_count[31:0]

STAT_TX_PACKET_1549_2047_BYTES_MSB: 0764

表 176: STAT_TX_PACKET_1549_2047_BYTES_MSB: 0764

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_1549_2047_bytes_count[48-1:32]

STAT_TX_PACKET_2048_4095_BYTES_LSB: 0768

表 177: STAT_TX_PACKET_2048_4095_BYTES_LSB: 0768

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_2048_4095_bytes_count[31:0]

STAT_TX_PACKET_2048_4095_BYTES_MSB: 076C

表 178: STAT_TX_PACKET_2048_4095_BYTES_MSB: 076C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_2048_4095_bytes_count[48-1:32]

STAT_TX_PACKET_4096_8191_BYTES_LSB: 0770

表 179: STAT_TX_PACKET_4096_8191_BYTES_LSB: 0770

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_4096_8191_bytes_count[31:0]

STAT_TX_PACKET_4096_8191_BYTES_MSB: 0774

表 180: STAT_TX_PACKET_4096_8191_BYTES_MSB: 0774

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_4096_8191_bytes_count[48-1:32]

STAT_TX_PACKET_8192_9215_BYTES_LSB: 0778

表 181: STAT_TX_PACKET_8192_9215_BYTES_LSB: 0778

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_8192_9215_bytes_count[31:0]

STAT_TX_PACKET_8192_9215_BYTES_MSB: 077C

表 182: STAT_TX_PACKET_8192_9215_BYTES_MSB: 077C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_8192_9215_bytes_count[48-1:32]

STAT_TX_PACKET_LARGE_LSB: 0780

表 183: STAT_TX_PACKET_LARGE_LSB: 0780

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_large_count[31:0]

STAT_TX_PACKET_LARGE_MSB: 0784

表 184: STAT_TX_PACKET_LARGE_MSB: 0784

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_large_count[48-1:32]

STAT_TX_PACKET_SMALL_LSB: 0788

表 185: STAT_TX_PACKET_SMALL_LSB: 0788

位	默认	类型	信号
31:0	0	HIST	stat_tx_packet_small_count[31:0]

STAT_TX_PACKET_SMALL_MSB: 078C

表 186: STAT_TX_PACKET_SMALL_MSB: 078C

位	默认	类型	信号
15:0	0	HIST	stat_tx_packet_small_count[48-1:32]

STAT_TX_BAD_FCS_LSB: 07B8

表 187: STAT_TX_BAD_FCS_LSB: 07B8

位	默认	类型	信号
31:0	0	HIST	stat_tx_bad_fcs_count[31:0]

STAT_TX_BAD_FCS_MSB: 07BC

表 188: STAT_TX_BAD_FCS_MSB: 07BC

位	默认	类型	信号
15:0	0	HIST	stat_tx_bad_fcs_count[48-1:32]

STAT_TX_UNICAST_LSB: 07D0

表 189: STAT_TX_UNICAST_LSB: 07D0

位	默认	类型	信号
31:0	0	HIST	stat_tx_unicast_count[31:0]

STAT_TX_UNICAST_MSB: 07D4

表 190: STAT_TX_UNICAST_MSB: 07D4

位	默认	类型	信号
15:0	0	HIST	stat_tx_unicast_count[48-1:32]

STAT_TX_MULTICAST_LSB: 07D8

表 191: STAT_TX_MULTICAST_LSB: 07D8

位	默认	类型	信号
31:0	0	HIST	stat_tx_multicast_count[31:0]

STAT_TX_MULTICAST_MSB: 07DC

表 192: STAT_TX_MULTICAST_MSB: 07DC

位	默认	类型	信号
15:0	0	HIST	stat_tx_multicast_count[48-1:32]

STAT_TX_BROADCAST_LSB: 07E0

表 193: STAT_TX_BROADCAST_LSB: 07E0

位	默认	类型	信号
31:0	0	HIST	stat_tx_broadcast_count[31:0]

STAT_TX_BROADCAST_MSB: 07E4

表 194: STAT_TX_BROADCAST_MSB: 07E4

位	默认	类型	信号
15:0	0	HIST	stat_tx_broadcast_count[48-1:32]

STAT_TX_VLAN_LSB: 07E8

表 195: STAT_TX_VLAN_LSB: 07E8

位	默认	类型	信号
31:0	0	HIST	stat_tx_vlan_count[31:0]

STAT_TX_VLAN_MSB: 07EC

表 196: STAT_TX_VLAN_MSB: 07EC

位	默认	类型	信号
15:0	0	HIST	stat_tx_vlan_count[48-1:32]

STAT_TX_PAUSE_LSB: 07F0

表 197: STAT_TX_PAUSE_LSB: 07F0

位	默认	类型	信号
31:0	0	HIST	stat_tx_pause_count[31:0]

STAT_TX_PAUSE_MSB: 07F4

表 198: STAT_TX_PAUSE_MSB: 07F4

位	默认	类型	信号
15:0	0	HIST	stat_tx_pause_count[48-1:32]

STAT_TX_USER_PAUSE_LSB: 07F8

表 199: STAT_TX_USER_PAUSE_LSB: 07F8

位	默认	类型	信号
31:0	0	HIST	stat_tx_user_pause_count[31:0]

STAT_TX_USER_PAUSE_MSB: 07FC

表 200: STAT_TX_USER_PAUSE_MSB: 07FC

位	默认	类型	信号
15:0	0	HIST	stat_tx_user_pause_count[48-1:32]

STAT_RX_TOTAL_PACKETS_LSB: 0808

表 201: STAT_RX_TOTAL_PACKETS_LSB: 0808

位	默认	类型	信号
31:0	0	HIST	stat_rx_total_packets_count[31:0]

STAT_RX_TOTAL_PACKETS_MSB: 080C

表 202: STAT_RX_TOTAL_PACKETS_MSB: 080C

位	默认	类型	信号
15:0	0	HIST	stat_rx_total_packets_count[48-1:32]

STAT_RX_TOTAL_GOOD_PACKETS_LSB: 0810

表 203: STAT_RX_TOTAL_GOOD_PACKETS_LSB: 0810

位	默认	类型	信号
31:0	0	HIST	stat_rx_total_good_packets_count[31:0]

STAT_RX_TOTAL_GOOD_PACKETS_MSB: 0814

表 204: STAT_RX_TOTAL_GOOD_PACKETS_MSB: 0814

位	默认	类型	信号
15:0	0	HIST	stat_rx_total_good_packets_count[48-1:32]

STAT_RX_TOTAL_BYTES_LSB: 0818

表 205: STAT_RX_TOTAL_BYTES_LSB: 0818

位	默认	类型	信号
31:0	0	HIST	stat_rx_total_bytes_count[31:0]

STAT_RX_TOTAL_BYTES_MSB: 081C

表 206: STAT_RX_TOTAL_BYTES_MSB: 081C

位	默认	类型	信号
15:0	0	HIST	stat_rx_total_bytes_count[48-1:32]

STAT_RX_TOTAL_GOOD_BYTES_LSB: 0820

表 207: STAT_RX_TOTAL_GOOD_BYTES_LSB: 0820

位	默认	类型	信号
31:0	0	HIST	stat_rx_total_good_bytes_count[31:0]

STAT_RX_TOTAL_GOOD_BYTES_MSB: 0824

表 208: STAT_RX_TOTAL_GOOD_BYTES_MSB: 0824

位	默认	类型	信号
15:0	0	HIST	stat_rx_total_good_bytes_count[48-1:32]

STAT_RX_PACKET_64_BYTES_LSB: 0828

表 209: STAT_RX_PACKET_64_BYTES_LSB: 0828

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_64_bytes_count[31:0]

STAT_RX_PACKET_64_BYTES_MSB: 082C

表 210: STAT_RX_PACKET_64_BYTES_MSB: 082C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_64_bytes_count[48-1:32]

STAT_RX_PACKET_65_127_BYTES_LSB: 0830

表 211: STAT_RX_PACKET_65_127_BYTES_LSB: 0830

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_65_127_bytes_count[31:0]

STAT_RX_PACKET_65_127_BYTES_MSB: 0834

表 212: STAT_RX_PACKET_65_127_BYTES_MSB: 0834

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_65_127_bytes_count[48-1:32]

STAT_RX_PACKET_128_255_BYTES_LSB: 0838

表 213: STAT_RX_PACKET_128_255_BYTES_LSB: 0838

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_128_255_bytes_count[31:0]

STAT_RX_PACKET_128_255_BYTES_MSB: 083C

表 214: STAT_RX_PACKET_128_255_BYTES_MSB: 083C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_128_255_bytes_count[48-1:32]

STAT_RX_PACKET_256_511_BYTES_LSB: 0840

表 215: STAT_RX_PACKET_256_511_BYTES_LSB: 0840

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_256_511_bytes_count[31:0]

STAT_RX_PACKET_256_511_BYTES_MSB: 0844

表 216: STAT_RX_PACKET_256_511_BYTES_MSB: 0844

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_256_511_bytes_count[48-1:32]

STAT_RX_PACKET_512_1023_BYTES_LSB: 0848

表 217: STAT_RX_PACKET_512_1023_BYTES_LSB: 0848

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_512_1023_bytes_count[31:0]

STAT_RX_PACKET_512_1023_BYTES_MSB: 084C

表 218: STAT_RX_PACKET_512_1023_BYTES_MSB: 084C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_512_1023_bytes_count[48-1:32]

STAT_RX_PACKET_1024_1518_BYTES_LSB: 0850

表 219: STAT_RX_PACKET_1024_1518_BYTES_LSB: 0850

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_1024_1518_bytes_count[31:0]

STAT_RX_PACKET_1024_1518_BYTES_MSB: 0854

表 220: STAT_RX_PACKET_1024_1518_BYTES_MSB: 0854

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_1024_1518_bytes_count[48-1:32]

STAT_RX_PACKET_1519_1522_BYTES_LSB: 0858

表 221: STAT_RX_PACKET_1519_1522_BYTES_LSB: 0858

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_1519_1522_bytes_count[31:0]

STAT_RX_PACKET_1519_1522_BYTES_MSB: 085C

表 222: STAT_RX_PACKET_1519_1522_BYTES_MSB: 085C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_1519_1522_bytes_count[48-1:32]

STAT_RX_PACKET_1523_1548_BYTES_LSB: 0860

表 223: STAT_RX_PACKET_1523_1548_BYTES_LSB: 0860

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_1523_1548_bytes_count[31:0]

STAT_RX_PACKET_1523_1548_BYTES_MSB: 0864

表 224: STAT_RX_PACKET_1523_1548_BYTES_MSB: 0864

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_1523_1548_bytes_count[48-1:32]

STAT_RX_PACKET_1549_2047_BYTES_LSB: 0868

表 225: STAT_RX_PACKET_1549_2047_BYTES_LSB: 0868

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_1549_2047_bytes_count[31:0]

STAT_RX_PACKET_1549_2047_BYTES_MSB: 086C

表 226: STAT_RX_PACKET_1549_2047_BYTES_MSB: 086C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_1549_2047_bytes_count[48-1:32]

STAT_RX_PACKET_2048_4095_BYTES_LSB: 0870

表 227: STAT_RX_PACKET_1549_2047_BYTES_MSB: 086C

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_2048_4095_bytes_count[31:0]

STAT_RX_PACKET_2048_4095_BYTES_MSB: 0874

表 228: STAT_RX_PACKET_2048_4095_BYTES_MSB: 0874

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_2048_4095_bytes_count[48-1:32]

STAT_RX_PACKET_4096_8191_BYTES_LSB: 0878

表 229: STAT_RX_PACKET_4096_8191_BYTES_LSB: 0878

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_4096_8191_bytes_count[31:0]

STAT_RX_PACKET_4096_8191_BYTES_MSB: 087C

表 230: STAT_RX_PACKET_4096_8191_BYTES_MSB: 087C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_4096_8191_bytes_count[48-1:32]

STAT_RX_PACKET_8192_9215_BYTES_LSB: 0880

表 231: STAT_RX_PACKET_8192_9215_BYTES_LSB: 0880

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_8192_9215_bytes_count[31:0]

STAT_RX_PACKET_8192_9215_BYTES_MSB: 0884

表 232: STAT_RX_PACKET_8192_9215_BYTES_MSB: 0884

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_8192_9215_bytes_count[48-1:32]

STAT_RX_PACKET_LARGE_LSB: 0888

表 233: STAT_RX_PACKET_LARGE_LSB: 0888

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_large_count[31:0]

STAT_RX_PACKET_LARGE_MSB: 088C

表 234: STAT_RX_PACKET_LARGE_MSB: 088C

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_large_count[48-1:32]

STAT_RX_PACKET_SMALL_LSB: 0890

表 235: STAT_RX_PACKET_SMALL_LSB: 0890

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_small_count[31:0]

STAT_RX_PACKET_SMALL_MSB: 0894

表 236: STAT_RX_PACKET_SMALL_MSB: 0894

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_small_count[48-1:32]

STAT_RX_UNDERSIZE_LSB: 0898

表 237: STAT_RX_UNDERSIZE_LSB: 0898

位	默认	类型	信号
31:0	0	HIST	stat_rx_undersize_count[31:0]

STAT_RX_UNDERSIZE_MSB: 089C

表 238: STAT_RX_UNDERSIZE_MSB: 089C

位	默认	类型	信号
15:0	0	HIST	stat_rx_undersize_count[48-1:32]

STAT_RX_FRAGMENT_LSB: 08A0

表 239: STAT_RX_FRAGMENT_LSB: 08A0

位	默认	类型	信号
31:0	0	HIST	stat_rx_fragment_count[31:0]

STAT_RX_FRAGMENT_MSB: 08A4

表 240: STAT_RX_FRAGMENT_MSB: 08A4

位	默认	类型	信号
15:0	0	HIST	stat_rx_fragment_count[48-1:32]

STAT_RX_OVERSIZE_LSB: 08A8

表 241: STAT_RX_OVERSIZE_LSB: 08A8

位	默认	类型	信号
31:0	0	HIST	stat_rx_oversize_count[31:0]

STAT_RX_OVERSIZE_MSB: 08AC

表 242: STAT_RX_OVERSIZE_MSB: 08AC

位	默认	类型	信号
15:0	0	HIST	stat_rx_oversize_count[48-1:32]

STAT_RX_TOOLONG_LSB: 08B0

表 243: STAT_RX_TOOLONG_LSB: 08B0

位	默认	类型	信号
31:0	0	HIST	stat_rx_toolong_count[31:0]

STAT_RX_TOOLONG_MSB: 08B4

表 244: STAT_RX_TOOLONG_MSB: 08B4

位	默认	类型	信号
15:0	0	HIST	stat_rx_toolong_count[48-1:32]

STAT_RX_JABBER_LSB: 08B8

表 245: STAT_RX_JABBER_LSB: 08B8

位	默认	类型	信号
31:0	0	HIST	stat_rx_jabber_count[31:0]

STAT_RX_JABBER_MSB: 08BC

表 246: STAT_RX_JABBER_MSB: 08BC

位	默认	类型	信号
15:0	0	HIST	stat_rx_jabber_count[48-1:32]

STAT_RX_BAD_FCS_LSB: 08C0

表 247: STAT_RX_BAD_FCS_LSB: 08C0

位	默认	类型	信号
31:0	0	HIST	stat_rx_bad_fcs_count[31:0]

STAT_RX_BAD_FCS_MSB: 08C4

表 248: STAT_RX_BAD_FCS_MSB: 08C4

位	默认	类型	信号
15:0	0	HIST	stat_rx_bad_fcs_count[48-1:32]

STAT_RX_PACKET_BAD_FCS_LSB: 08C8

表 249: STAT_RX_PACKET_BAD_FCS_LSB: 08C8

位	默认	类型	信号
31:0	0	HIST	stat_rx_packet_bad_fcs_count[31:0]

STAT_RX_PACKET_BAD_FCS_MSB: 08CC

表 250: STAT_RX_PACKET_BAD_FCS_MSB: 08CC

位	默认	类型	信号
15:0	0	HIST	stat_rx_packet_bad_fcs_count[48-1:32]

STAT_RX_STOMPED_FCS_LSB: 08D0

表 251: STAT_RX_STOMPED_FCS_LSB: 08D0

位	默认	类型	信号
31:0	0	HIST	stat_rx_stomped_fcs_count[31:0]

STAT_RX_STOMPED_FCS_MSB: 08D4

表 252: STAT_RX_STOMPED_FCS_MSB: 08D4

位	默认	类型	信号
15:0	0	HIST	stat_rx_stomped_fcs_count[48-1:32]

STAT_RX_UNICAST_LSB: 08D8

表 253: STAT_RX_UNICAST_LSB: 08D8

位	默认	类型	信号
31:0	0	HIST	stat_rx_unicast_count[31:0]

STAT_RX_UNICAST_MSB: 08DC

表 254: STAT_RX_UNICAST_MSB: 08DC

位	默认	类型	信号
15:0	0	HIST	stat_rx_unicast_count[48-1:32]

STAT_RX_MULTICAST_LSB: 08E0

表 255: STAT_RX_MULTICAST_LSB: 08E0

位	默认	类型	信号
31:0	0	HIST	stat_rx_multicast_count[31:0]

STAT_RX_MULTICAST_MSB: 08E4

表 256: STAT_RX_MULTICAST_MSB: 08E4

位	默认	类型	信号
15:0	0	HIST	stat_rx_multicast_count[48-1:32]

STAT_RX_BROADCAST_LSB: 08E8

表 257: STAT_RX_BROADCAST_LSB: 08E8

位	默认	类型	信号
31:0	0	HIST	stat_rx_broadcast_count[31:0]

STAT_RX_BROADCAST_MSB: 08EC

表 258: STAT_RX_BROADCAST_MSB: 08EC

位	默认	类型	信号
15:0	0	HIST	stat_rx_broadcast_count[48-1:32]

STAT_RX_VLAN_LSB: 08F0

表 259: STAT_RX_VLAN_LSB: 08F0

位	默认	类型	信号
31:0	0	HIST	stat_rx_vlan_count[31:0]

STAT_RX_VLAN_MSB: 08F4

表 260: STAT_RX_VLAN_MSB: 08F4

位	默认	类型	信号
15:0	0	HIST	stat_rx_vlan_count[48-1:32]

STAT_RX_PAUSE_LSB: 08F8

表 261: STAT_RX_PAUSE_LSB: 08F8

位	默认	类型	信号
31:0	0	HIST	stat_rx_pause_count[31:0]

STAT_RX_PAUSE_MSB: 08FC

表 262: STAT_RX_PAUSE_MSB: 08FC

位	默认	类型	信号
15:0	0	HIST	stat_rx_pause_count[48-1:32]

STAT_RX_USER_PAUSE_LSB: 0900

表 263: STAT_RX_USER_PAUSE_LSB: 0900

位	默认	类型	信号
31:0	0	HIST	stat_rx_user_pause_count[31:0]

STAT_RX_USER_PAUSE_MSB: 0904

表 264: STAT_RX_USER_PAUSE_MSB: 0904

位	默认	类型	信号
15:0	0	HIST	stat_rx_user_pause_count[48-1:32]

STAT_RX_INRANGEERR_LSB: 0908

表 265: STAT_RX_INRANGEERR_LSB: 0908

位	默认	类型	信号
31:0	0	HIST	stat_rx_inrangeerr_count[31:0]

STAT_RX_INRANGEERR_MSB: 090C

表 266: STAT_RX_INRANGEERR_MSB: 090C

位	默认	类型	信号
15:0	0	HIST	stat_rx_inrangeerr_count[48-1:32]

STAT_RX_TRUNCATED_LSB: 0910

表 267: STAT_RX_TRUNCATED_LSB: 0910

位	默认	类型	信号
31:0	0	HIST	stat_rx_truncated_count[31:0]

STAT_RX_TRUNCATED_MSB: 0914

表 268: STAT_RX_TRUNCATED_MSB: 0914

位	默认	类型	信号
15:0	0	HIST	stat_rx_truncated_count[48-1:32]

STAT_RX_TEST_PATTERN_MISMATCH_LSB: 0918

表 269: STAT_RX_TEST_PATTERN_MISMATCH_LSB: 0918

位	默认	类型	信号
31:0	0	HIST	stat_rx_test_pattern_mismatch_count[31:0]

STAT_RX_TEST_PATTERN_MISMATCH_MSB: 091C

表 270: STAT_RX_TEST_PATTERN_MISMATCH_MSB: 091C

位	默认	类型	信号
15:0	0	HIST	stat_rx_test_pattern_mismatch_count[48-1:32]

STAT_FEC_INC_CORRECT_COUNT_LSB: 0920

表 271: STAT_FEC_INC_CORRECT_COUNT_LSB: 0920

位	默认	类型	信号
31:0	0	HIST	stat_fec_inc_correct_count_count[31:0]

STAT_FEC_INC_CORRECT_COUNT_MSB: 0924

表 272: STAT_FEC_INC_CORRECT_COUNT_MSB: 0924

位	默认	类型	信号
15:0	0	HIST	stat_fec_inc_correct_count_count[48-1:32]

STAT_FEC_INC_CANT_CORRECT_COUNT_LSB: 0928

表 273: STAT_FEC_INC_CANT_CORRECT_COUNT_LSB: 0928

位	默认	类型	信号
31:0	0	HIST	stat_fec_inc_cant_correct_count_count[31:0]

STAT_FEC_INC_CANT_CORRECT_COUNT_MSB: 092C

表 274: STAT_FEC_INC_CANT_CORRECT_COUNT_MSB: 092C

位	默认	类型	信号
15:0	0	HIST	stat_fec_inc_cant_correct_count_count[48-1:32]

利用子系统进行设计

本节包含有关利用该核来协助简化设计的指导信息和其它信息。

通用设计指南

使用设计示例

由 Vivado 设计工具所创建的 40G/50G High Speed Ethernet 子系统的每个实例交付时均包含设计示例，此设计示例可先在器件中实现，然后再对其进行仿真。您可将此设计作为起点来开展自己的设计，或者也可在遇到困难时将其用于对您的应用进行完整性检查。请参阅“设计示例”内容，以获取有关将设计示例用于子系统以及设计示例自定义的信息。

了解设计难度

在任何技术中实现赛灵思 40G/50G High Speed Ethernet 子系统设计皆非易事，其难度也进一步受以下因素影响：

- 最高系统时钟频率
- 目标器件架构
- 应用的性质

所有 40G/50G High Speed Ethernet 子系统实现都需要仔细考量系统性能要求。流水打拍、逻辑映射、布局约束和逻辑复制都是有助于提升系统性能的方法。

寄存信号

为了在可编程器件设计中简化时序并提升系统性能，请使用户应用与子系统之间的所有输入和输出保持处于已寄存状态。这意味着来自用户应用的所有输入和输出都应来自于或者连接至触发器。虽然可能并非所有路径都能寄存信号，但这样可简化时序分析，便于赛灵思工具对设计进行布局布线。

识别时序关键信号

随设计示例所提供的约束可识别应应用的关键信号和时序约束。

仅限执行允许的修改

不得修改子系统。任何修改都可能对系统时序和协议合规性造成负面影响。只能通过在生成子系统时，选择“Customization IP”对话框中的选项来执行受支持的子系统用户配置。

时钟设置

本节描述了组件支持封装文件层级上的所有 40G/50G 配置的时钟设置。其中包含 3 种完全不同的时钟架构（按功能和选项区分）：

- 仅限 PCS/PMA 时钟设置
- 含 PCS/PMA 时钟设置的 40G/50G MAC
- 含 PCS/PMA 时钟设置的低时延 40G/50G MAC

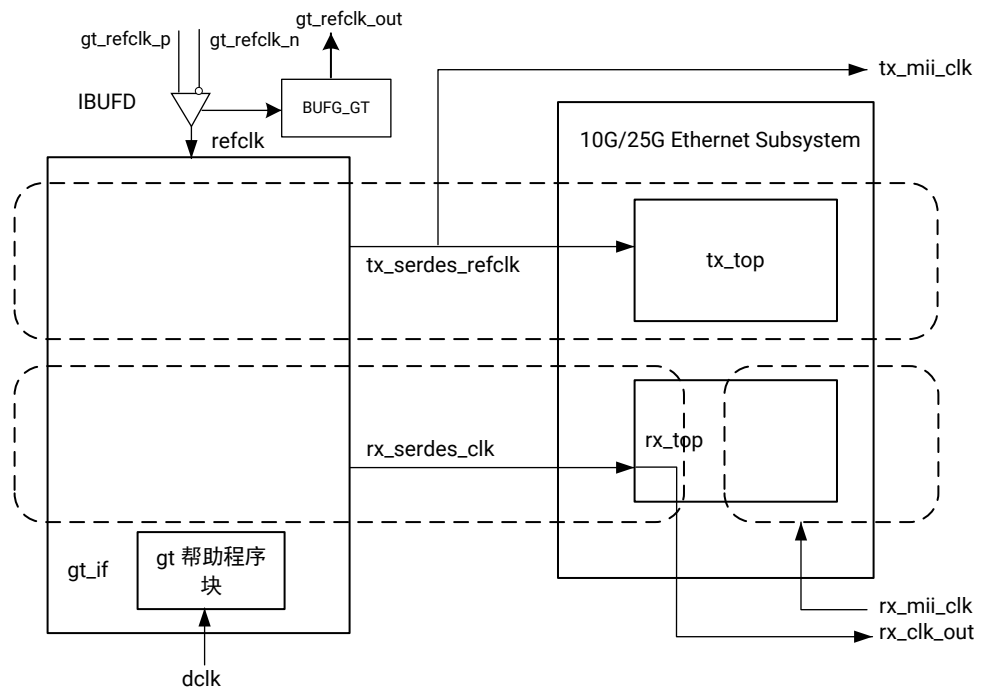
另外，还描述了 [自动协商和链路训练时钟](#)。

注释：在 Vivado® IDE 中所选“Data Path Interface”为 256 位常规 AXI4-Stream 时，TX/RX AXI4-Stream 设计示例应使用 `tx_out_clk`；否则可能导致包不匹配。

仅限 PCS/PMA 时钟设置

下图展示了 40G/50G PCS 的时钟架构。数据路径中有 3 个时钟域，如下图中虚线所示：

图 20：PCS/PMA 时钟设置



X15168-101122

refclk_p0、refclk_n0 和 tx_serdes_refclk

`refclk` 差分对是必需的，用作为 FPGA 的输入。设计示例包含缓冲器，用于将该时钟转换为单端信号 `refclk`，此信号用作为 GT 块的参考时钟。`tx_serdes_refclk` 直接衍生自 `refclk`。请注意，必须选择 `refclk`，以确保 `tx_mii_clk` 满足 802.3 的要求，即范围应分别在 312.5 MHz 的 100 ppm 内（针对 40G）和 390.625 MHz 的 100 ppm 内（针对 50G）。

tx_mii_clk

tx_mii_clk 是输出，与 tx_serdes_refclk 相同。整个 TX 路径由此时钟驱动。您必须将 TX 路径 mii 总线同步到此时钟输出。所有 TX 控制信号和状态信号均以此时钟为参考。

rx_serdes_clk

rx_serdes_clk 衍生自 GT 块中的输入数据串流。输入数据串流由此时钟域中的 RX 核进行处理。

rx_clk_out

提供的 rx_clk_out 输出信号可供 RX 核所处理的 RX 控制信号和状态信号作为参考。其频率与 rx_serdes_clk 相同。

rx_mii_clk

rx_mii_clk 输入必须同步到 RX XLGMII/50GMII 数据总线。此时钟和 RX XLGMII/50GMII 总线必须在所需频率（即 312.5 MHz (40G) 和 390.625 MHz (50G)）的 100 ppm 内。

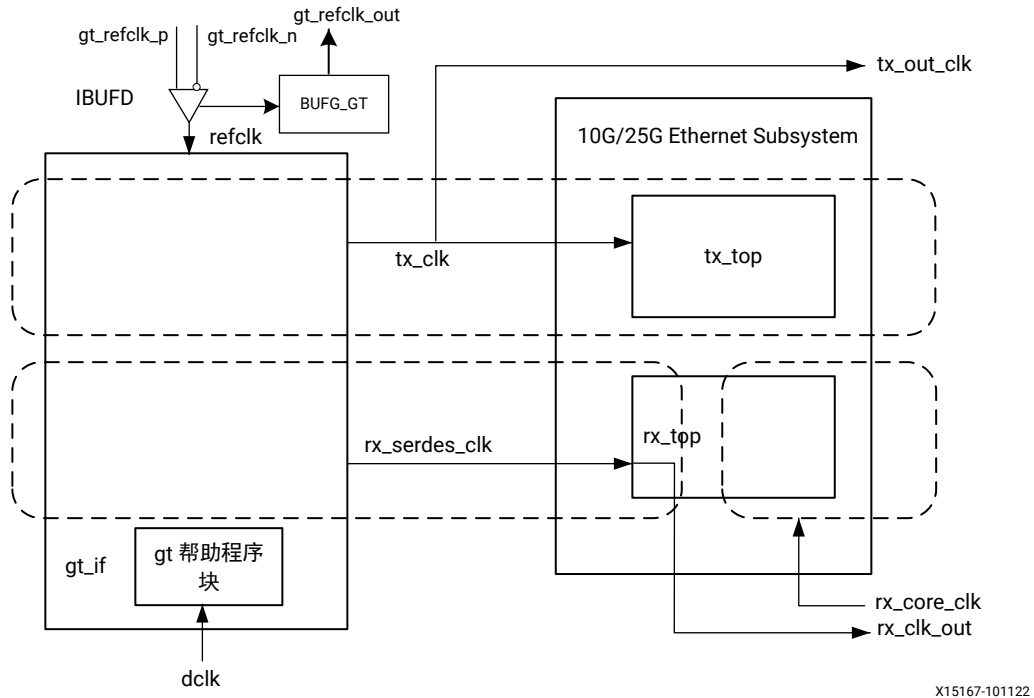
dclk

dclk 信号必须为稳定且便于使用的时钟。它用作为 GT 帮助程序块（用于启动 GT 本身）的参考频率。在设计示例中，典型值为 75 MHz，该值直接衍生自 VCU107 评估板上可用的 300 MHz 时钟。请注意，GT 帮助程序块必须已知实际频率才能正常运行。

含 PCS/PMA 时钟设置的 40G/50G MAC

下图展示了含 PCS/PMA 时钟设置的 40/50G MAC 的时钟架构。此版本的子系统在 RX 中包含 FIFO。在数据路径中有 3 个时钟域，如下图中虚线所示：

图 21：含 PCS/PMA 时钟设置的 40G/50G MAC



X15167-101122

refclk_p0、refclk_n0 和 tx_serdes_refclk

refclk 差分对是必需的，用作为 FPGA 的输入。设计示例包含缓冲器，用于将该时钟转换为单端信号 refclk，此信号用作为 GT 块的参考时钟。tx_serdes_refclk 直接衍生自 refclk。请注意，必须选择 refclk，以确保 tx_serdes_refclk 满足 802.3 的要求，即范围应分别在 312.5 MHz 的 100 ppm 内（针对 40G）和 390.625 MHz 的 100 ppm 内（针对 50G）。

tx_clk_out

此时钟用作为时钟设置数据输入 TX AXI4-Stream 接口，它还用作为 TX 控制信号和状态信号的参考时钟。其频率与 tx_serdes_refclk 相同。

rx_clk_out

提供的 rx_clk_out 输出信号可供 RX 核所处理的 RX 控制信号和状态信号作为参考。其频率与 rx_serdes_clk 相同。

rx_clk

rx_clk 是 RX 核的输入时钟。该 rx_clk 可用作 rx_core_clk（必须通过设计示例进行驱动）。应以与 tx_clk 相同的频率来驱动 rx_core_clk。启用 FIFO 时，系统端数据路径的首选运行模式是将 tx_clk_out 连接到 rx_core_clk。通过这种方式进行连接时，RX AXI4-Stream 接口和 TX AXI4-Stream 接口都位于同一时钟域中。

dclk

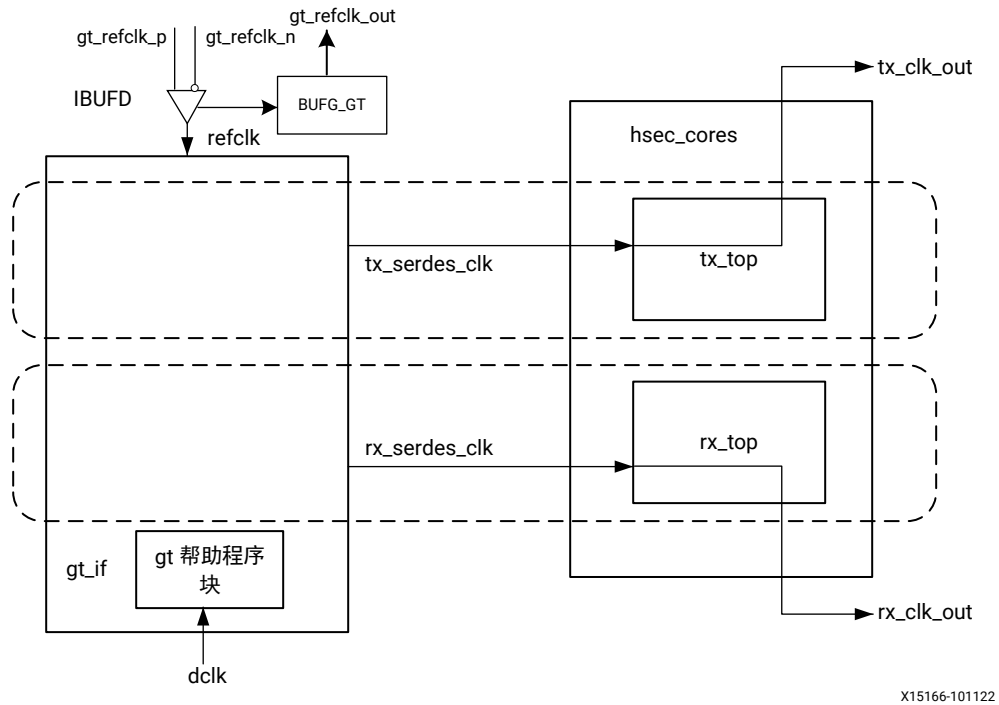
dclk 信号必须为稳定且便于使用的时钟。它用作为 GT 帮助程序块（用于启动 GT 本身）的参考频率。在设计示例中，典型值为 75 MHz，该值直接衍生自 VCU107 评估板上可用的 300 MHz 时钟。

注释：GT 帮助程序块必须已知实际频率才能正常运行。

含 PCS/PMA 时钟设置的低时延 40G/50G MAC

含 PCS/PMA 时钟设置的低时延 40G/50G MAC 的时钟架构如下图所示。低时延是通过省略 RX FIFO 来实现的，这导致时钟排布产生变化。在数据路径中有 2 个时钟域，如下图虚线所示。

图 22：含 PCS/PMA 时钟设置的低时延 40G/50G MAC



X15166-101122

refclk_p0、refclk_n0 和 tx_serdes_refclk

refclk 差分对是必需的，用作为 FPGA 的输入。设计示例包含缓冲器，用于将该时钟转换为单端信号 refclk，此信号用作为 GT 块的参考时钟。tx_serdes_refclk 直接衍生自 refclk。请注意，必须选择 refclk 以确保 tx_serdes_refclk 满足 802.3 的要求，即范围应分别在 312.5 MHz 的 100 ppm 内（针对 40G）和 390.625 MHz 的 100 ppm 内（针对 50G）。

tx_clk_out

此时钟用作为时钟设置数据输入 TX AXI4-Stream 接口，它还用作为 TX 控制信号和状态信号的参考时钟。其频率与 tx_serdes_refclk 相同。由于不存在 TX FIFO，因此您必须立即响应 tx_axis_tready 信号。

rx_clk_out

提供的 rx_clk_out 输出信号可供 RX 核所处理的 RX 控制信号和状态信号作为参考。其频率与 rx_serdes_clk 相同。由于不存在 RX FIFO，因此该时钟还用于驱动 RX AXI4-Stream 接口。按此排布方式，rx_clk_out 与 tx_clk_out 的频率不同，且彼此间不存在定义的相位关系。

dclk

dclk 信号必须为稳定且便于使用的时钟。它作为 GT 帮助程序块（用于启动 GT 本身）的参考频率。在设计示例中，典型值为 75 MHz，该值直接衍生自 VCU107 评估板上可用的 300 MHz 时钟。请注意，GT 帮助程序块必须已知实际频率才能正常运行。

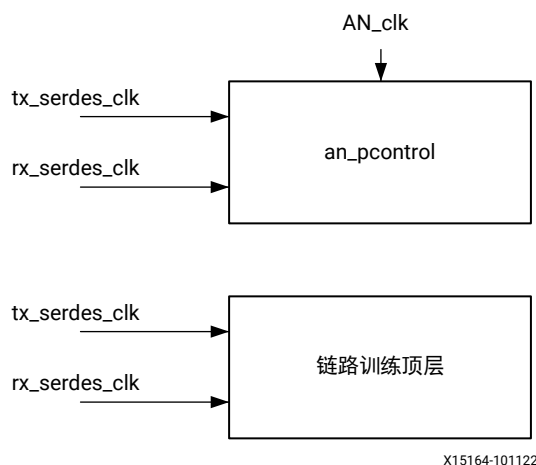
自动协商和链路训练时钟

下图显示了自动协商和链路训练块的时钟架构。

注释：除非选择了 50GBASE-KR 或 50GBASE-CR 特性，否则不包含这些块。

自动协商和链路训练块与 MAC 和 PCS 分开运作，因此处于不同的时钟域。

图 23：自动协商和链路训练时钟



tx_serdes_clk

tx_serdes_clk 驱动用于自动协商和链路训练的 TX 线一侧的逻辑。DME 帧在此时钟域上生成。

rx_serdes_clk

rx_serdes_clk 驱动用于自动协商和链路训练的 RX 线一侧的逻辑。

AN_clk

AN_clk 驱动自动协商状态机。所有功能信号都位于此时钟域。AN_clk 可以是任何便于使用的频率。在设计示例中，AN_clk 连接到 dclk 输入，其典型频率为 75 MHz。自动协商状态机必须已知 AN_clk 频率，因为这是所有定时器的参考频率。

LogiCORE 设计示例时钟设置和复位

40G/50G High Speed Ethernet 子系统具有用于 RX 和 TX 路径的独立复位输入，这些路径可以单独断言有效。在 RX 和 TX 路径内，不同时钟域各自都有复位。复位过程很简单，唯一要求是对应时钟不稳定时复位必须保持断言有效。

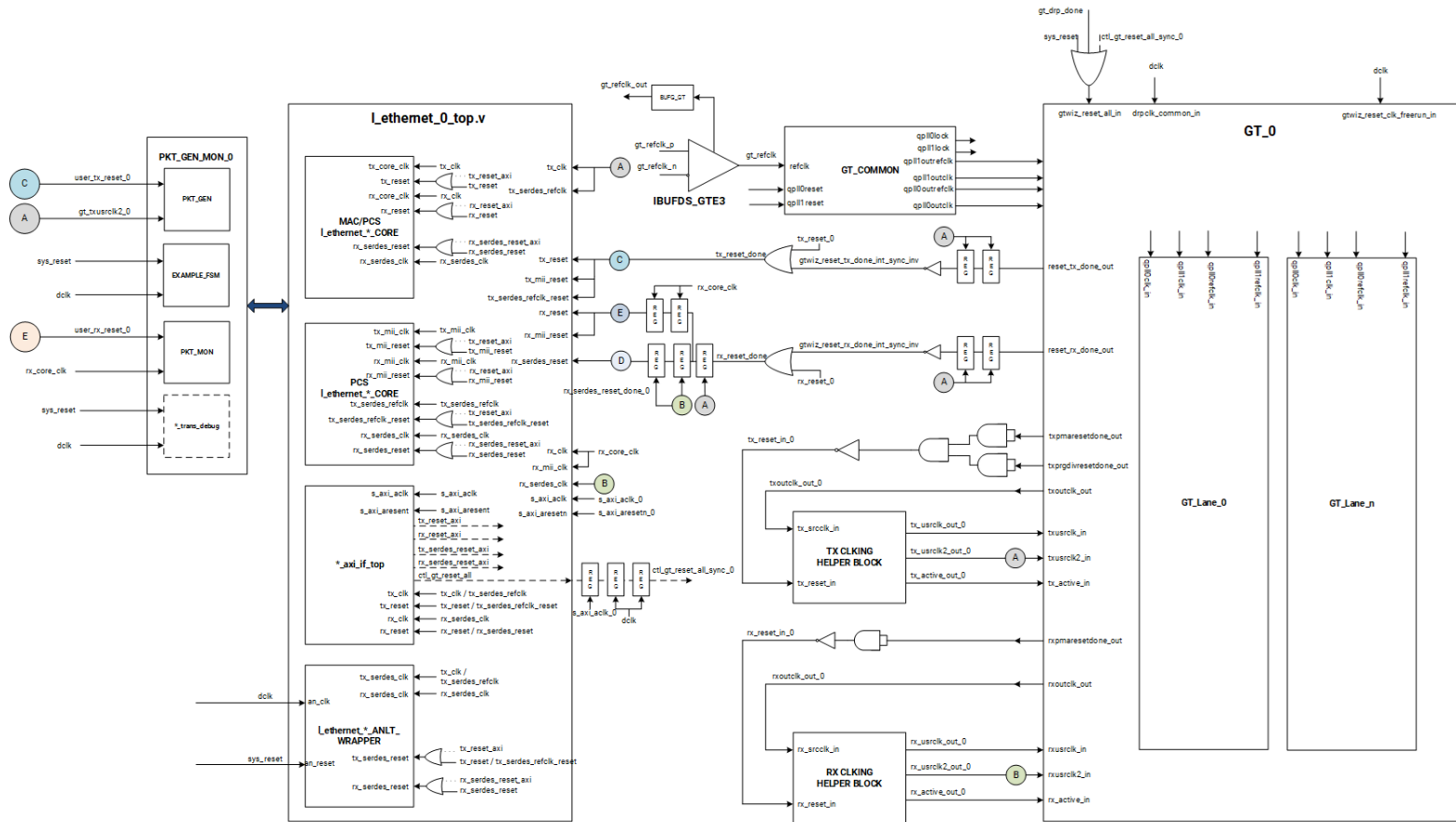
40G/50G High Speed Ethernet 子系统负责确保不同复位在内部正确彼此交互且接口工作正常（即，不同复位断言有效/断言无效在顺序方面没有要求）。而您负责确保复位保持不变，直至对应时钟达到完全稳定状态为止。

注释：40G/50G High Speed Ethernet 子系统的某些控制输入仅限在该核保持复位状态时才能进行修改。如果其中某一输入需要更改，则相应的 RX 或 TX AXI4-Stream 复位输入（`rx_reset` 或 `tx_reset`）必须断言有效，直至控制输入达到稳定状态为止。有关这些输入的列表，请参阅表 2-2。目前，所有复位与对应的时钟保持同步。也就是说，复位断言高电平有效时，对应时钟上必须有 0-1 转换，这样才能执行复位。

下图演示了您使用 Vivado 工具实现设计示例时的时钟设置和复位结构。

适用于 UltraScale/UltraScale+ 器件的单核 (1x50G) - 异步时钟模式 (GTY) (2x25G)

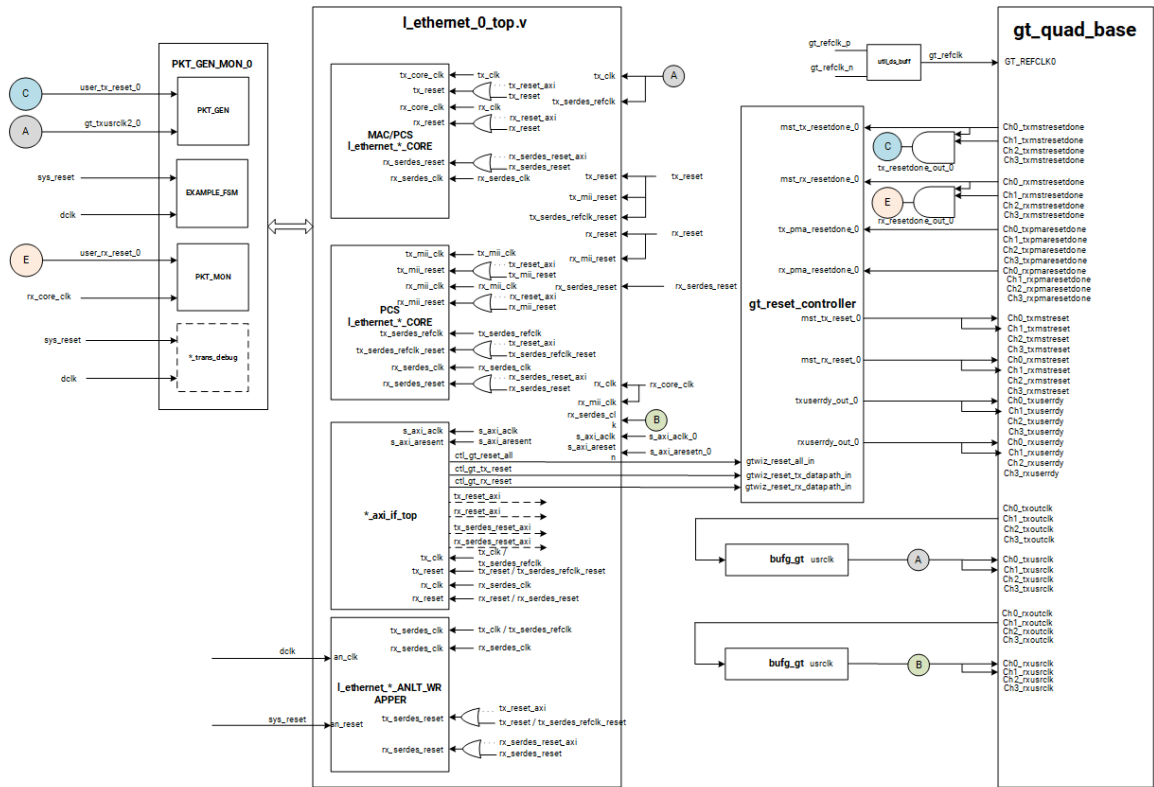
图 24：适用于 UltraScale/UltraScale+ 器件的单核 (1x50G) - 异步时钟模式 (GTY) (2x25G)



374629-000720

适用于 Versal ACAP 的单核 (1x50G) - 异步时钟模式 (GTY) (2x25G)

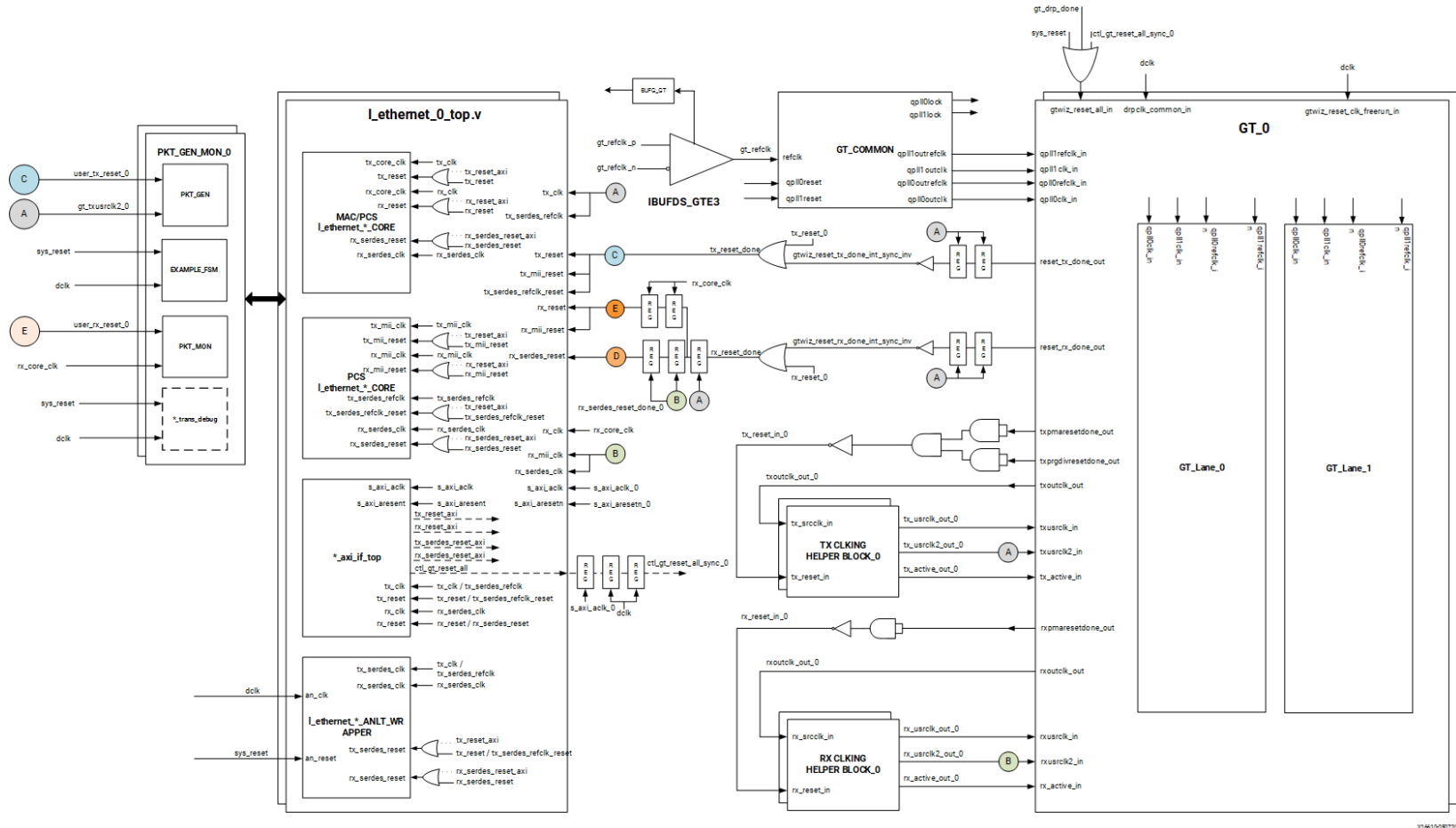
图 25：适用于 Versal ACAP 的单核 (1x50G) - GTY (2x25G)



32931-08720

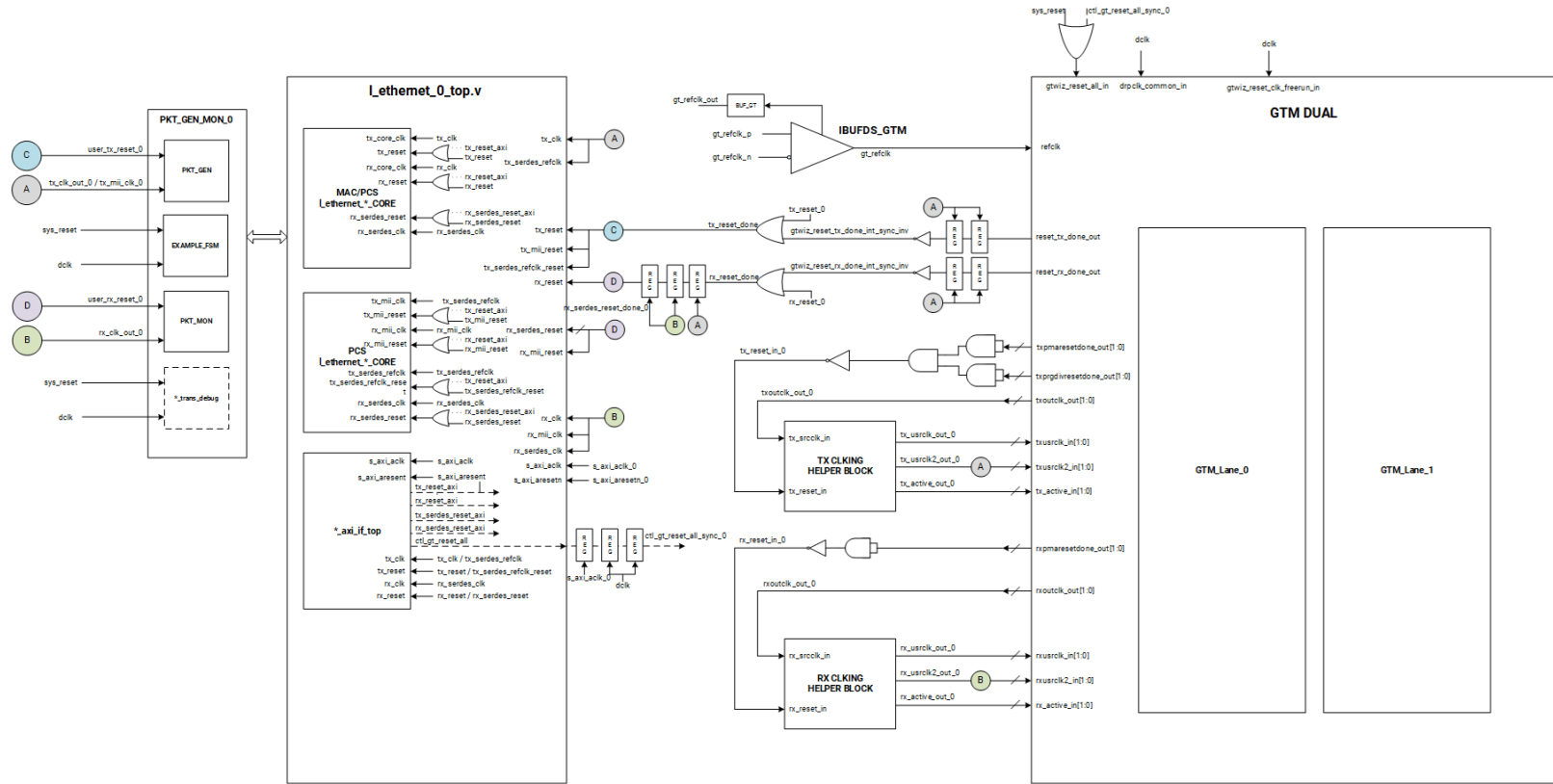
适用于 UltraScale/UltraScale+ 器件的多核 (2x50G) - 异步时钟模式 GTY (2x25G)

图 27：适用于 UltraScale/UltraScale+ 器件的多核 (2x50G) - 异步时钟模式 GTY (2x25G)



适用于 UltraScale+ 的单核 (1x50G) - 异步时钟模式 GTM (1x50G)

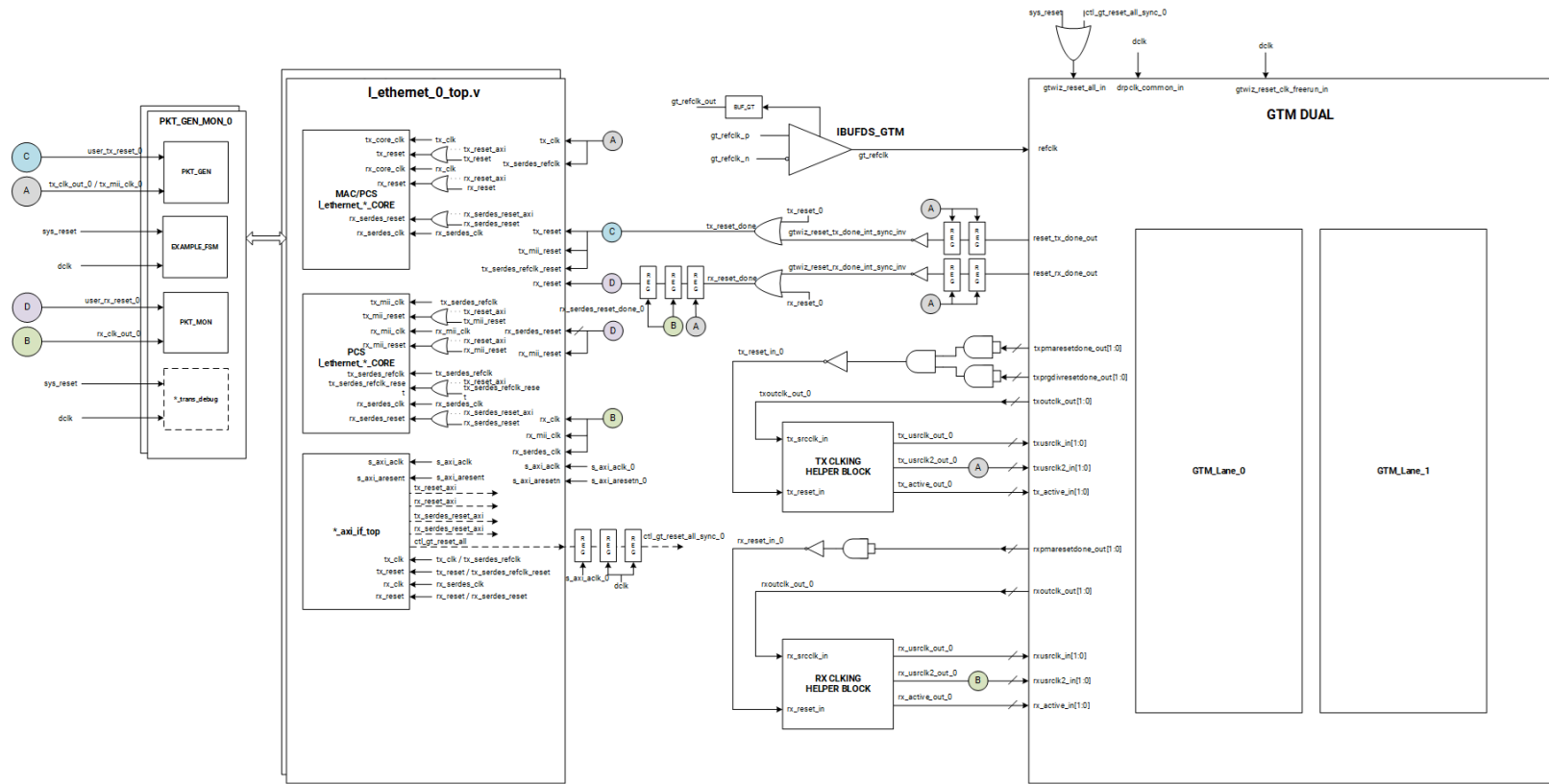
图 28：适用于 UltraScale+ 的单核 (1x50G) - 异步时钟模式 GTM (1x50G)



12270-04020

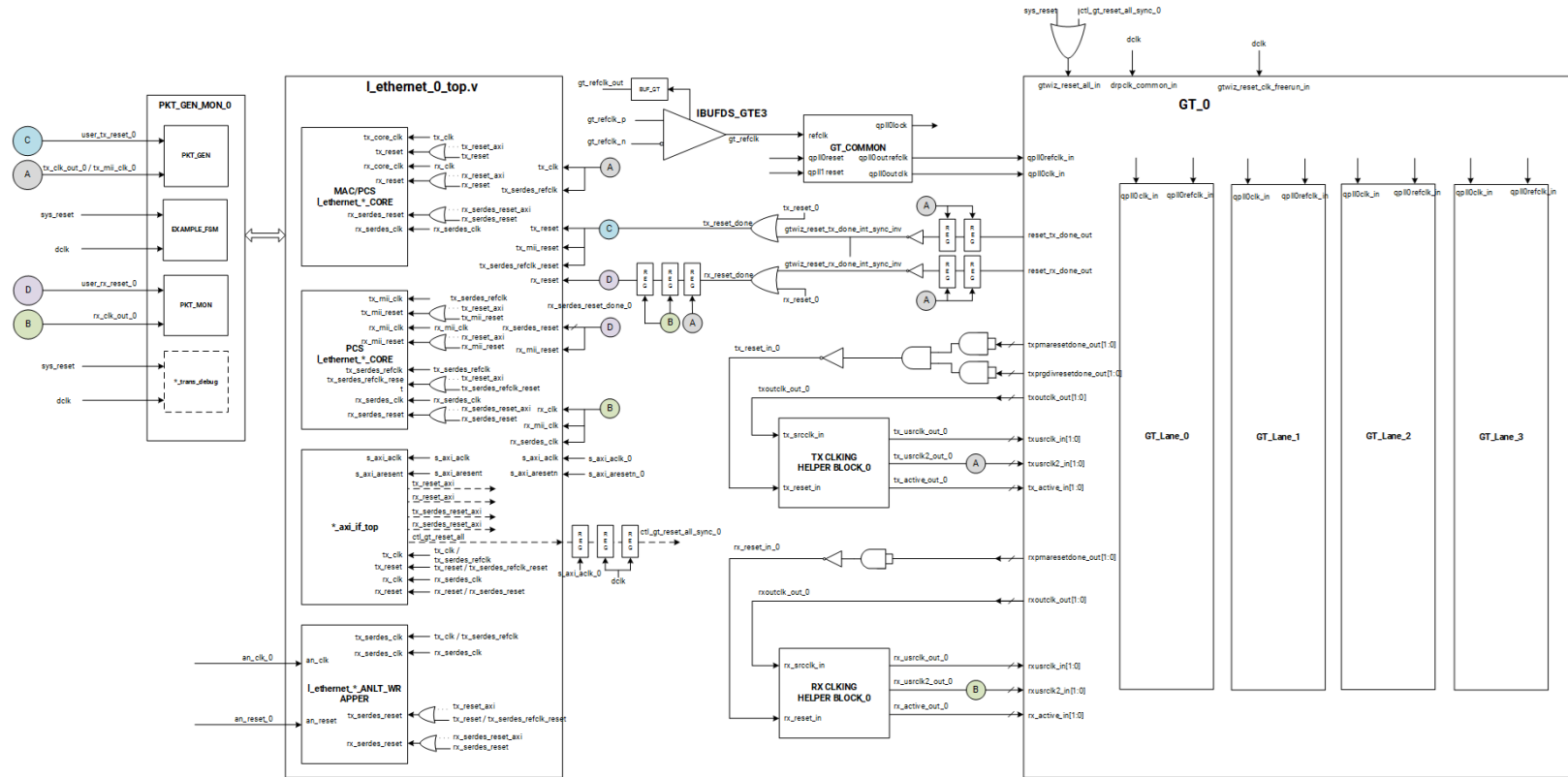
适用于 UltraScale+ 的多核 (2x50G) - 异步时钟模式 GTM (2x50G)

图 29：适用于 UltraScale+ 的多核 (2x50G) - 异步时钟模式 GTM (2x50G)



单核 (1x40G) - 异步时钟模式 GTY/GTH (4x10G)

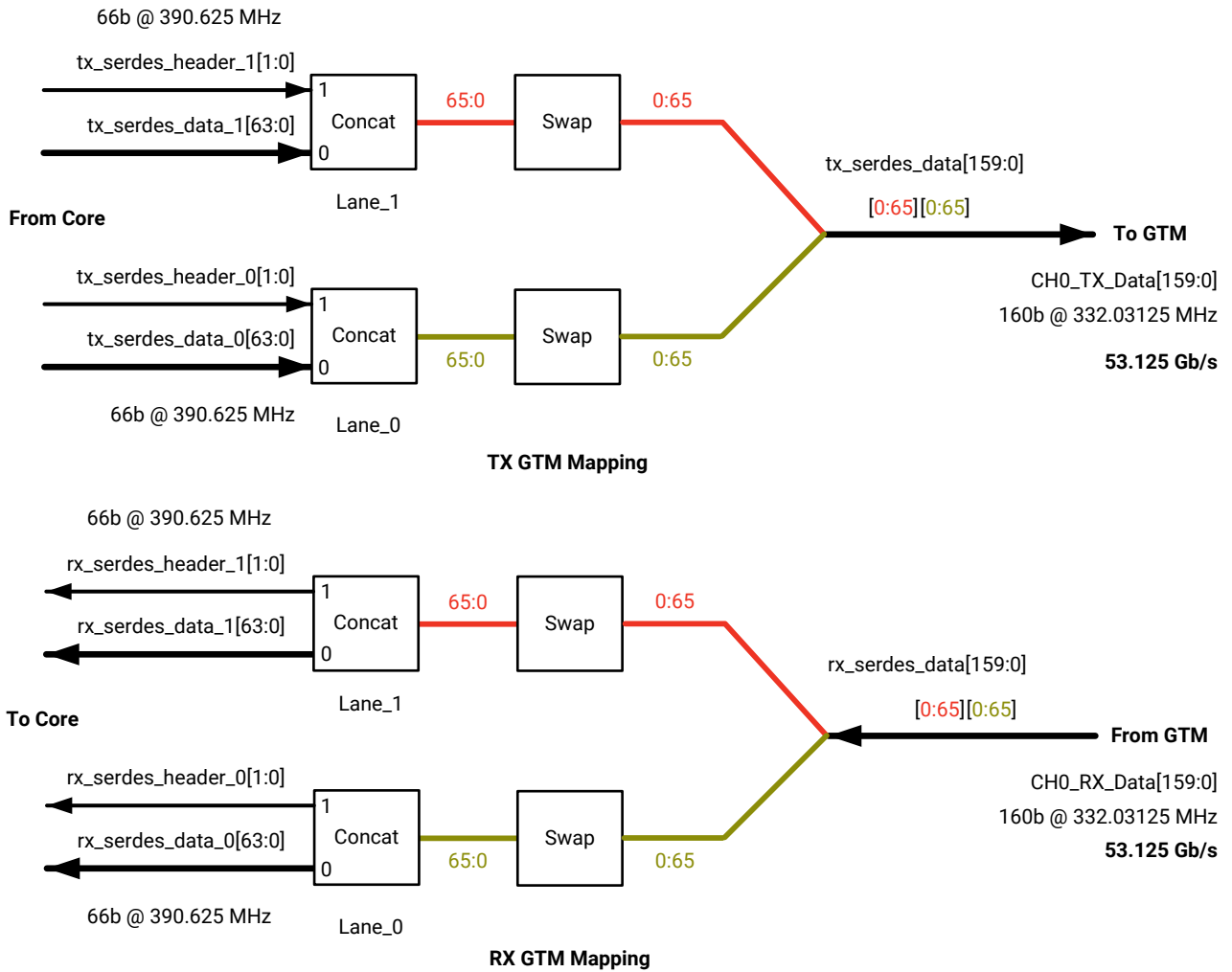
图 30: 单核 (1x40G) - 异步时钟模式 GTY/GTH (4x10G)



1027208720

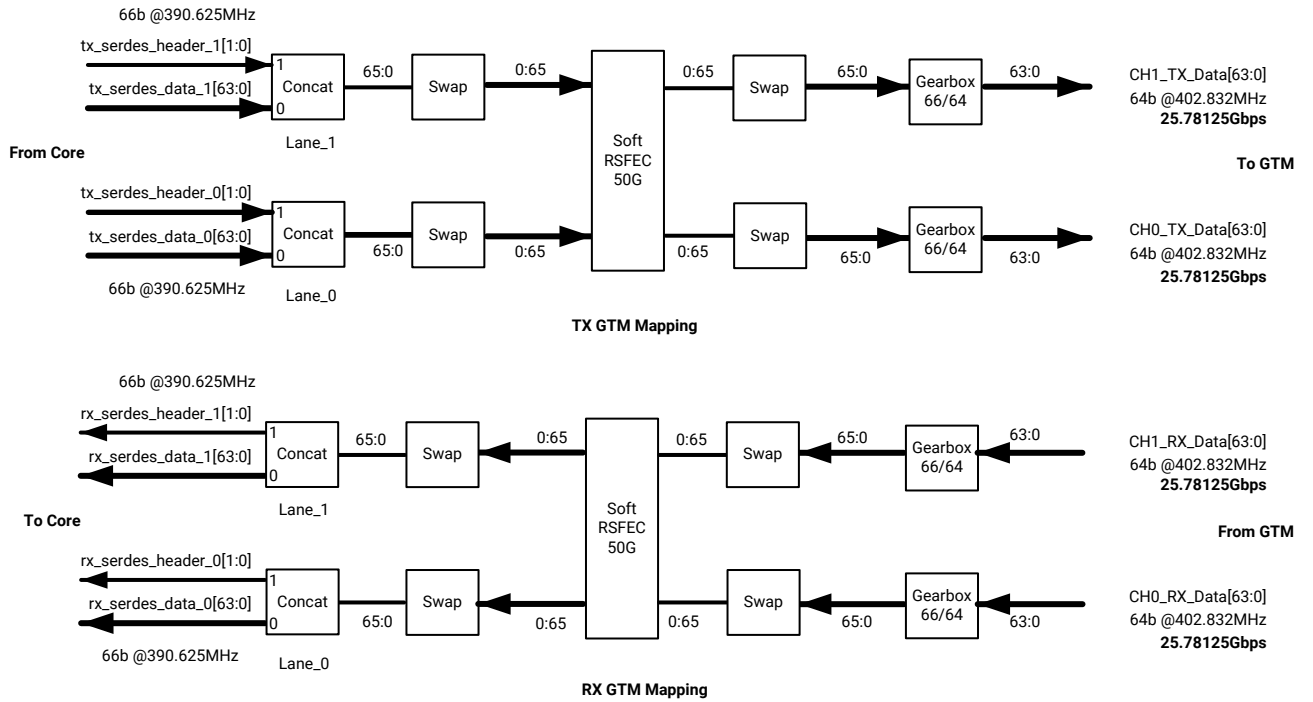
适用于 GTM 的 SerDes 数据映射

图 31: 50GAUI1 (含 GTM 集成硬核 RS-FEC)



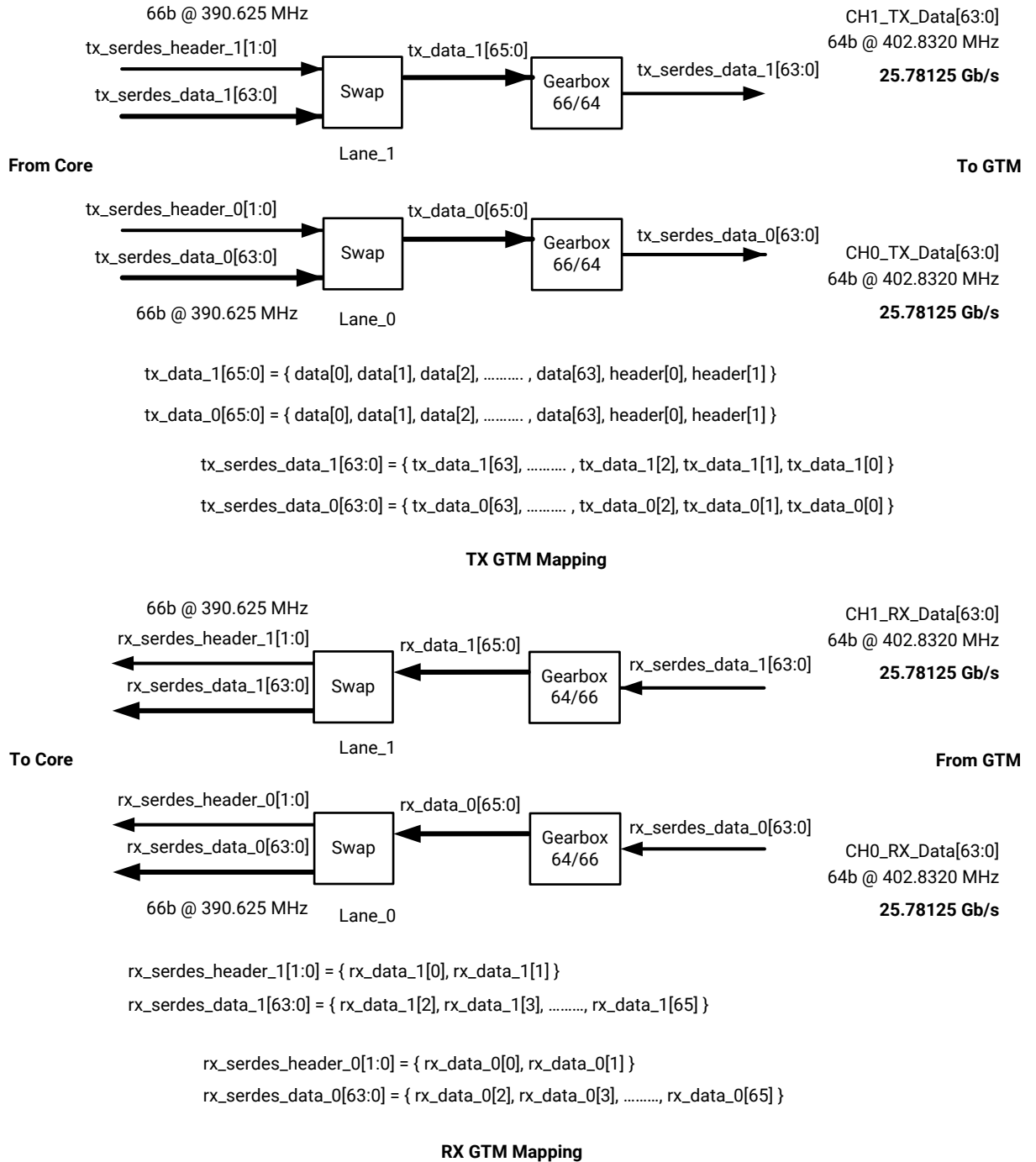
X22030-080621

图 32: LAUI2 (含软核 RS-FEC)



X22034-080621

图 33: LAUI2 (不含软核 RS-FEC)



X22033-080621

表 275: 受支持的 40G/50G - GTM 配置

协议	通道宽度	线速率	编码	SerDes 宽度
50GAUI-1 ¹	x1	53.125 Gb/s	PAM4	160b

表 275：受支持的 40G/50G - GTM 配置 (续)

协议	通道宽度	线速率	编码	SerDes 宽度
LAUI-2 (含软核 RSFEC)	x2	25.78125 Gb/s	NRZ	64b
LAUI-2 (不含 RSFEC)	x2	25.78125 Gb/s	NRZ	64b
XLAUI-1 ²	x4	10.3125 Gb/s	NRZ	64b
50GAUI-2 (含软核 RSFEC)	x2	26.5625 Gb/s	NRZ	64b

注释：

1. 使用单一 GTM 双通道的 2 条通道，还可提供 2 个核选项以供使用。
2. XLAUI-1 使用 2 个 GTM 双通道。

适用于 40G/50G 子系统的 IEEE PTP 1588v2

概述

本节详述了包含 MAC 层时 40G/50G Ethernet Subsystem 的数据包时间戳功能。时间戳选项必须在从 IP 目录生成子系统时或者对 IP 核进行异步排序时指定。此功能当前支持单步和双步 IEEE PTP 1588 功能。在本附录中描述了单步操作以供参考。

以太网帧在入口和出口均生成时间戳。该选项可用于实现所有种类的 IEEE 1588 时钟，包括：“Ordinary Clock”（普通时钟）、“Transparent Clock”（透明时钟）和“Boundary Clock”（边界时钟）。它还可用于在系统的入口端口和出口端口处为数据包生成通用时间戳。虽然此功能可用于各种数据包时间戳应用，但本附录其余部分假定您将同时实现 IEEE 1588 精确时间协议 (PTP)。

IEEE 1588 定义了用于在整个网络中执行时序同步的协议。每个 1588 网络都具有单一主时钟参考，通常它是通过最佳主时钟算法来选定的。该主时钟会定期对其系统定时器参考计数器进行采样，并使用定义的数据包格式在整个网络中发射采样所得的时间值。该定时器应在 1588 时序包开始发射时进行采样（生成 1 个时间戳）。因此，要在整个网络上实现高精度同步，就需要精确的时间戳。如果将此采样所得定时器值（时间戳）放入触发时间戳的包内，那么这就被称为单步操作。或者，也可将此时间戳值置于后续数据包内，称为双步操作。

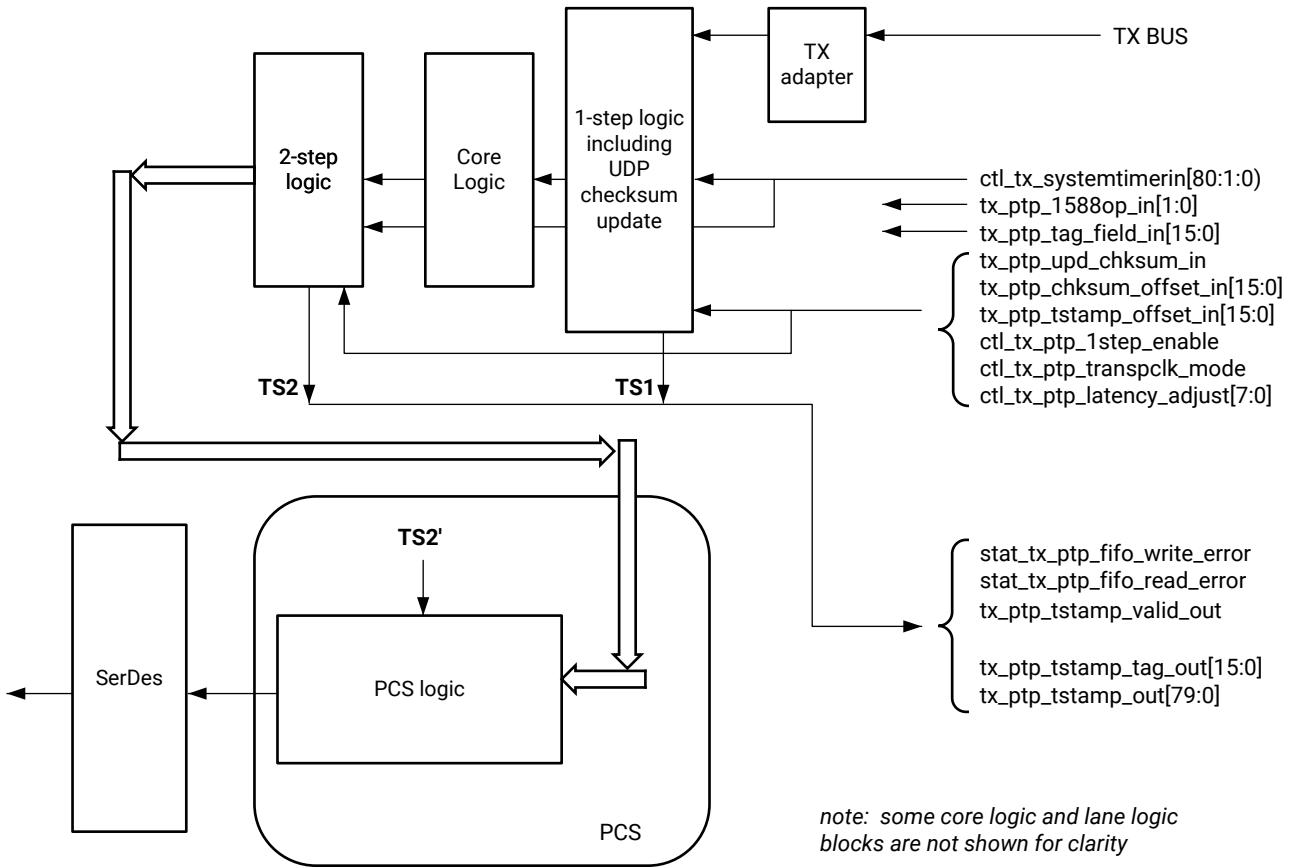
网络上的其它时序从器件会从网络时序主时钟接收到这些时序参考包，并尝试据此同步自己的本地定时器参考时钟。此机制要求在接收到 1588 时序包时，这些以太网端口同样生成时间戳（对其自己的本地定时器进行采样）。有关 1588 操作方式的进一步解释不在本文档的讨论范围内。假定读者阅读本节其余部分时已熟悉 IEEE 1588 规范相关内容。

提供给子系统的 1588 定时器以及从该定时器生成的后续时间戳可通过 2 种格式提供，在子系统生成时选择所用格式。

- 当日时间 (ToD) 格式：IEEE 1588-2008 格式由无符号的 48 位秒字段和 32 位纳秒字段组成。
- 纠正字段 (Correction Field) 格式：IEEE 1588-2008 数字格式包含 1 个 64 位有符号字段，该字段表示纳秒数乘以 2^{16} （请参阅 IEEE 1588 第 13.3.2.7 条）。此定时器应从 0 开始向上计数（全量程）直至 $2^{64}-1$ 为止，然后再回绕。

出口

图 34：出口



note: some core logic and lane logic blocks are not shown for clarity

X16339-080621

如图所示，根据所需的是单步操作还是双步操作，时间戳逻辑存在于两处不同位置。单步操作需用户数据报协议 (UDP) 校验和及 FCS 更新，因此使用 FCS 核逻辑。

TS 参考定义如下：

- TS1：选中单步操作时的输出时间戳信号。
- TS2：选中双步操作时的输出时间戳信号。
- TS2'：2 个时间戳纠正到该平面。

对于 TS2，始终应用纠正以使其可供 TS2' 平面作为参考。TS1 可根据 `ctl_tx_ptp_latency_adjust[10:0]` 信号的值来选择是否应用 TS2' 纠正。此信号的默认值是在生成子系统时确定的。

以下是时延调整值。

- 50G 普通时钟 = 450
- 50G 透明时钟 = 549
- 40G 普通时钟 = 605
- 40G 透明时钟 = 663

- 40G 256 位常规 AXI4-Stream 普通时钟 = 620
- 40G 256 位常规 AXI4-Stream 透明时钟 = 725

在发射端，“Command” 字段由客户端随发送的帧（以供发射之用）并行提供给子系统。这表示根据每个帧来确定要执行的 1588 功能（即，无操作、单步操作或双步操作），并且对于单步帧，这还可用于指示是否存在需更新的 UDP 校验和字段。

如果使用 ToD 格式，那么对于单步操作和双步操作，完整捕获的 80 位 ToD 时间戳将使用“1588v2 端口列表和描述”表中所定义的附加端口返回给客户端逻辑。如果使用“Correction Field”（纠正字段）格式，那么对于单步操作和双步操作，完整捕获的 64 位时间戳将使用“1588v2 端口列表和描述”表中定义的附加端口返回给客户端逻辑（根据表中定义，上位数据设置为 0）。

如果使用 ToD 格式，那么对于单步操作，完整捕获的 80 位 ToD 时间戳将插入帧中。如果使用“Correction Field”格式，那么对于单步操作，捕获的 64 位时间戳将与帧中包含的现有“Correction Field”求和，求和所得结果将覆盖到帧中的原“Correction Field”中。

针对单步时间戳支持的帧类型包括：

- 原始以太网 (Raw Ethernet)
- UDP/IPv4
- UDP/IPv6

对于单步 UDP 帧类型，将根据 IETF RFC 1624 来更新 UDP 校验和。对于所有单步帧，将在完成所有帧修改之后计算“Ethernet Frame Check Sequence (FCS)”（以太网帧校验序列）字段。对于双步发射操作，支持所有“Precision Time Protocol (PTP)”（精确时间协议）帧类型。

出口时间戳功能的操作模式由 1588 命令端口上的设置判定。命令端口中所含信息用于指示以下任一结果：

- 无操作：该帧并非 PTP 帧，不应执行任何时间戳操作。
- 需执行双步操作并在“Command”字段中提供 1 个标签值（用户序列 ID）；对于该帧，应设置时间戳，并且所设时间戳可供客户端逻辑使用，所提供的标签值则可供该帧使用。附加 MAC 发射器端口可提供此功能。
- 需执行单步操作
 - 对于 ToD 定时器和时间戳格式，将在命令端口中提供时间戳偏移；应对该帧设置时间戳，并且此时间戳应按提供的偏移（字节数）插入帧中。
 - 对于“Correction Field”格式，将在命令端口中提供“Correction Field”偏移值；应对该帧设置时间戳，捕获的 64 位时间戳应与该帧中包含的现有“Correction Field”求和，求和所得结果将覆盖到帧中的原“Correction Field”中。

对于帧修改后的单步操作，还应更新/重新计算帧的 CRC 值。对于 UDP IPv4 格式和 IPv6 PTP 格式的帧，需更新/重新计算该帧的报头中的校验和值。

对于单步 UDP 帧类型，将根据 IETF RFC 1624 来更新 UDP 校验和。

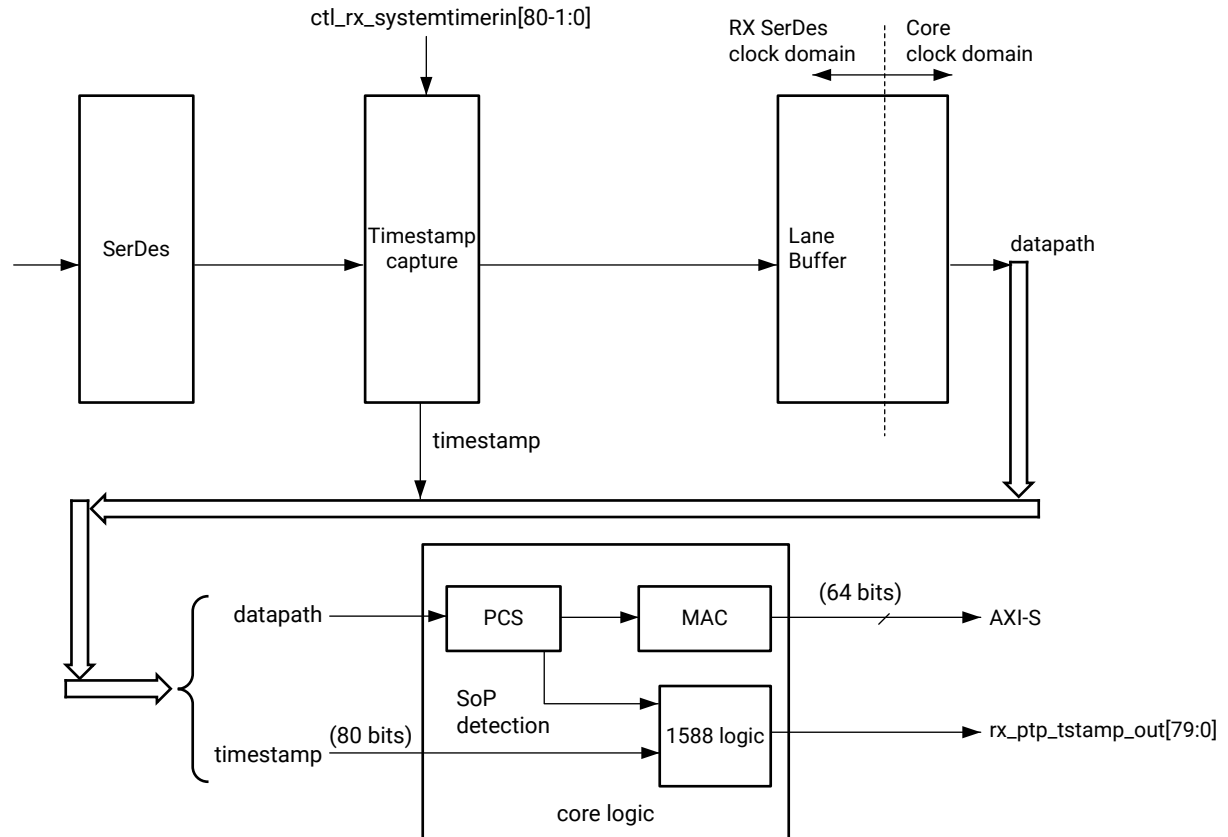
- 如果使用“ToD”格式，为了能正确执行此更新功能，计算发送（以供发射）的帧的原校验和值时，针对时间戳数据应使用 0 值。使用“Correction Field”格式时，则不适用此特殊限制。
- 如果使用“Correction Field”格式，则适用另一种限制；1588 PTP 帧报头中的“UDP Checksum”（UDP 校验和）字段与“Correction Field”之差为固定间隔（以字节为单位），以支持 1588 PTP 帧定义。此要求用于最大限度缩短通过 MAC 的时延，因为校验和与纠正字段都必须完全包含在 MAC 流水线内才能正确更新校验和。此特殊限制不适用于“ToD”格式，因为计算所得原时间戳数据为 0 值；因此，校验和与时间戳位置可分别位于同一帧中的不同位置。

相关信息

[1588v2 端口列表和描述](#)

入口

图 35：入口



X16340-080621

入口逻辑不会通过解析入口数据包来搜索 1588 (PTP) 帧。而是改为提取接收到的每个帧的时间戳，并将该值输出到用户逻辑。该功能始终启用，但如果无需此功能，则可忽略时间戳输出。

先过滤 PCS 解码器，然后再过滤时间戳，这样即可仅保留对应于 SOP 的时间戳。这些 80 位时间戳在系统侧输出。在 SOP 周期内且 `ena_out = 1` 时，此时间戳有效。

1588v2 端口列表和描述

下表描述了包含数据包时间戳功能的情况下存在的其它信号。

表 276：1588v2 端口列表和描述

信号	I/O	描述	时钟域
IEEE 1588 接口 - TX 路径			
ctl_tx_systemtimerin[80-1:0]	输入	适用于 TX 的系统定时器输入。在正常时钟模式下，由 32 个 LSB 承载纳秒 (nsec)，由 48 个 MSB 承载秒 (second)。在透明时钟模式下，位 62:16 用于承载纳秒，位 15:0 则用于承载小数纳秒。请参阅 IEEE 1588v2 以了解相关表示法的定义。 此输入必须位于 TX SerDes 时钟域内。	tx_serdes_clk
tx_ptp_tstamp_valid_out	输出	此位用于表示 TX 系统接口上当前呈现的有效时间戳。	tx_clk_out
tx_ptp_tstamp_tag_out[15:0]	输出	对应于 tx_ptp_tag_field_in[15:0] 的标签输出	tx_clk_out
tx_ptp_tstamp_out[80-1:0]	输出	已发射的包 SOP 的时间戳，对应于它通过捕获平面的时间。用于双步 1588 操作。时间格式与定时器输入相同。	tx_clk_out
tx_ptp_1588op_in[1:0]	输入	此信号应在数据包的首个周期内有效，并适用于 AXI4-Stream 发射接口上呈现的每个包。 2'b00 - No operation: 表示无操作，不记录时间戳，不修改帧。 2'b01 - 1-step: 表示单步，应记录时间戳并将其插入帧。 2'b10 - 2-step: 表示双步，应记录时间戳并使用对应双步操作的额外端口将该时间戳返回到客户端。帧本身无需修改。 2'b11 - Reserved: 表示保留，相当于“无操作”。	tx_clk_out
ctl_tx_ptp_1step_enable	输入	设为 1 时，该位即启用单步操作。 注释： 仅当对应的复位输入断言有效时，才应更改此输入。包发射期间，它不应动态更改。	tx_clk_out
ctl_ptp_transpclk_mode	输入	设为 1 时，此输入会将时间戳逻辑置于透明时钟模式下。在此模式下，系统定时器输入将解释为纠正值。TX 将根据 IEEE 1588v2 中定义的进程，把该纠正值添加到 TX 时间戳。 这样对应的传入 PTP 包纠正字段应按适当的 RX 时间戳完成调整。 注释： 仅当对应的复位输入断言有效时，才应更改此输入。包发射期间，它不应动态更改。	tx_clk_out
rx_ptp_tstamp_valid_out	输出	此位用于表示 RS 系统接口上当前呈现的有效时间戳。 注释： 此项仅针对 256 位常规串流接口数据路径有效。	
tx_ptp_tag_field_in[15:0]	输入	该字段的使用取决于 1588 操作。此信号应在数据包的首个周期内有效。 <ul style="list-style-type: none"> · 对于“No operation”，将忽略该字段。 · 对于“1-step”和“2-step”，此字段将成为标签字段。该标签值将通过使用双步操作的额外端口，与对应当前帧的时间戳一起返回至客户端。该标签值可供软件用于确保该时间戳能够与其发送的 PTP 帧（用于发射）相匹配。 	tx_clk_out
ctl_tx_ptp_latency_adjust[10:0]	输入	该总线可用于调整与双步时间戳有关的单步 TX 时间戳。位 [10:3] 的单位为纳秒，而位 [2:0] 的单位则为小数纳秒。 注释： 仅当对应的复位输入断言有效时，才应更改此输入。包发射期间，它不应动态更改。	tx_clk_out

表 276: 1588v2 端口列表和描述 (续)

信号	I/O	描述	时钟域
stat_tx_ptp_fifo_write_error	输出	发射 PTP FIFO 写入错误。此状态值为 1 表示 PTP Tag 写入期间发生错误。需 TX 路径复位才能清除此错误。	tx_clk_out
stat_tx_ptp_fifo_read_error	输出	发射 PTP FIFO 读取错误。此状态值为 1 表示 PTP Tag 读取期间发生错误。需 TX 路径复位才能清除此错误。	tx_clk_out
IEEE 1588 接口 - RX 路径			
ctl_rx_systemtimerin[80-1:0]	输入	适用于 RX 的系统定时器输入。时间格式与 TX 相同。此输入必须与 RX SerDes 位于相同时钟域内。	rx_serdes_clk
rx_ptp_tstamp_out[80-1:0]	输出	已接收的包 SOP 的时间戳，对应于它通过捕获平面的时间。此信号将在数据包的首个周期内有效。	rx_clk_out
tx_ptp_upd_chksum_in	输入	请参阅“IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号中的 tx_ptp_upd_chksum_in”。	
tx_ptp_pclane_out	输出	请参阅“IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号中的 tx_ptp_pclane_out”。	
tx_ptp_chksum_offset_in	输入	请参阅“IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号中的 tx_ptp_chksum_offset_in”。	
rx_ptp_pclane_out	输出	请参阅“IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号中的 rx_ptp_pclane_out”。	
rx_lane_aligner_fill	输出	请参阅“IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号中的 rx_lane_aligner_fill”。	

相关信息

[IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号](#)

IEEE PTP 1588v2 功能描述

40G/50G 子系统的 IEEE 1588 功能在硬件级别为出入双向提供准确的以太网帧时间戳。

时间戳根据前面定义的输入时钟源（系统定时器）来捕获。不过，此时间源必须与 SerDes 位于同一个时钟域中。您可能需要使用外部电路来重新定时。

在典型的应用中，PTP 算法（或 servo，不属于此 IP）将移除一段时间内（许多包样本）的时间戳错误。这种错误最好要尽可能小，以最大程度减少收敛时间和从时钟漂移。PTP 包通常每秒钟发射约 10 次。

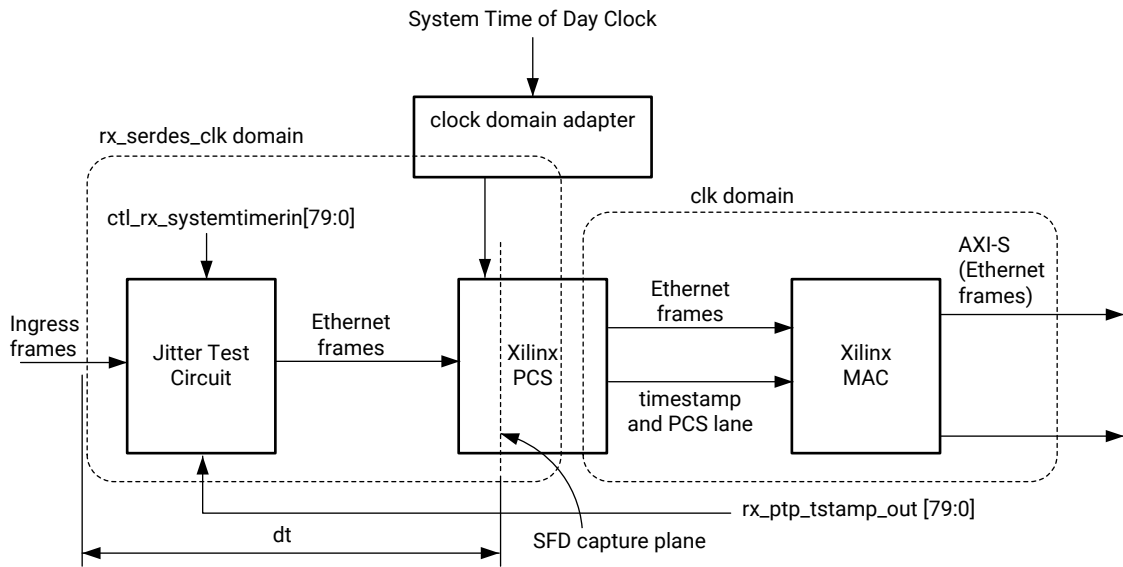
所有入口帧都会获得时间戳。由您负责解读收到的帧，并确定特定帧是否含有 PTP 信息（通过其以太类型来判断）以及判断此时间戳需保留还是丢弃。

如果出口帧被标记为 PTP 帧，则为其设置时间戳。出口帧的时间戳与其用户提供的标签相匹配。

传入帧的时间戳则按照与包起始相对应的 AXI4-Stream 周期，并行显示在用户接口上。然后，您可以根据需要将时间戳添加到包上。

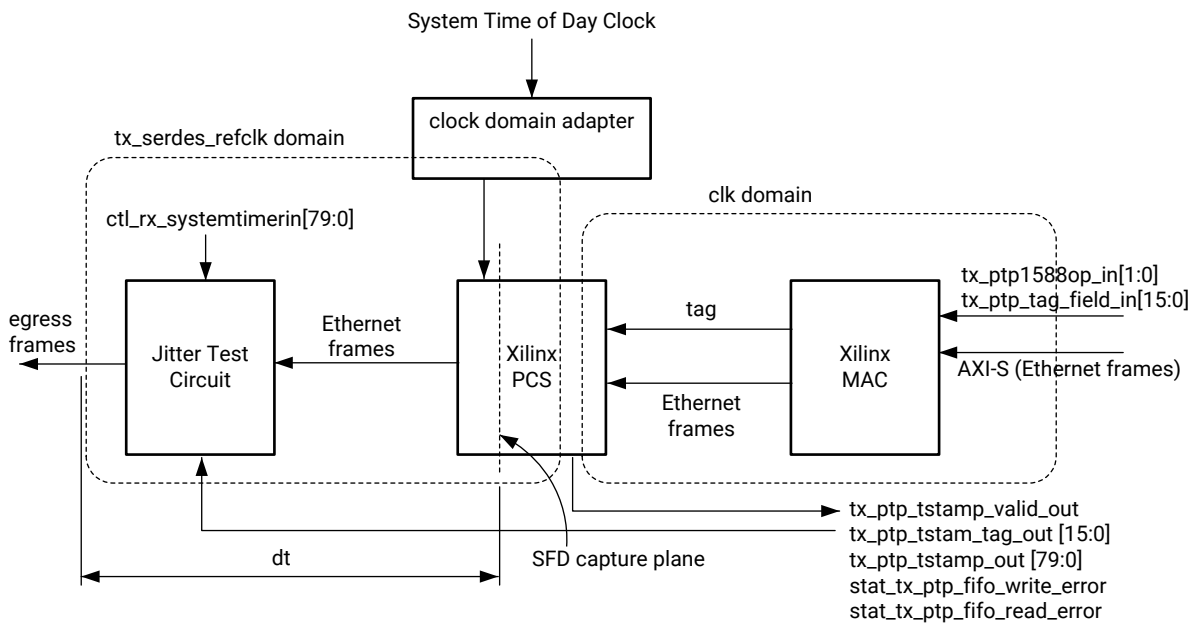
按照定义，时间戳的捕获时间与 SOP 通过 40G/50G High Speed Ethernet 子系统中捕获平面的时间一致，具体如下方板级原理图中所示：

图 36：接收



X16341-080621

图 37：发射



X16342-080621

性能

时间戳逻辑的性能经过测试，如先前引用的图例所示。在 RX 侧，抖动测试电路会在帧起始进入测试电路的同时记录系统定时器。稍后，此时间戳将被 RX PCS 捕获，并最终在系统侧 AXI4-Stream 接口 (rx_ptp_tstamp_out[79:0]) 上输出。这 2 次时间捕获之差的变化量 dt 即定义为时间戳逻辑的“抖动”性能。对于 tx_ptp_tstamp_out[79:0]，TX 测试与此类似。

理论上，40G/50G 子系统时间戳逻辑能判定的跨 SOP 捕获平面的时间应在 80 位系统定时器输入的粒度范围内。因此，如果系统定时器的时间段为 1 nsec，那么时间戳的精度将在 1 nsec 抖动范围内。1 nsec 同时也是 IEEE 1588 定义的 80 位字段的最低有效位的粒度。

实际上，还有其它因素会限制真实系统中可实现的精度。

时钟域

实际上，系统定时器输入的精度应与 SerDes 时钟粒度相同。因此，系统定时器输入的时钟域交汇也应一并纳入考量。例如，如果 SerDes 时钟频率为 390 MHz，那么系统定时器实际粒度为 2.56 ns，这也是捕获时间戳的时钟的粒度。因此，可预计存在额外的 2.56 ns 变化。

收发器

在数据路径中添加 SerDes 并不会影响 40G/50G 子系统的抖动性能，但可能导致不对称。

在 1588 时钟应用中，RX + TX SerDes 时延会成为回路延迟的一部分，因此由 1588 协议来执行测量。为了最大程度提升从时钟的精度，最好将回路不对称纳入考量（RX 与 TX SerDes 时延之差）。赛灵思可以为您的器件专用的各种 SerDes 设置提供收发器时延。如果要将对非对称纳入您的 PTP 系统考量范围，则需要联系数据路径中其它收发器的供应商以了解其特性。

UltraScale™ 和 UltraScale+™ 收发器能够报告 RX 时延。RX 收发器变化主要源于其内部弹性缓冲器的填充级别。请参阅收发器指南以获取更多详细信息。

前向纠错

在时间戳捕获的线路侧会发生前向纠错（对应第 74 条和第 91 条）。因此，添加 FEC 将不会影响 40G/50G 子系统的时间戳捕获的精度。与前述 SerDes 情况相似，FEC 的额外时延总量（RX + TX）也将由 1588 协议来进行测量。

（请注意，SOP 在发射的 FEC 帧中保持不可见状态，直至由 RX FEC 功能将其解码后才可见。）

为了最大程度提升 1588 从时钟的精度，最好明确 RX 和 TX 方向中 FEC 时延的非对称性。请联系赛灵思以了解特定 FEC 的 RX 和 TX 时延及其配置。如果 PTP 系统并非赛灵思 FEC 实现，那么您可能还需要从该系统中的链路伙伴供应商处获取此信息。

接收偏差纠正

在多通道系统（如 40G 和 50G）中，对应于 SOP 的包可能发生在任何通道内。此外，通道彼此间可能存在相对偏差。40G/50G 子系统能够通过报告 SOP 通道及其偏差，将 SOP 帧的到达通道纳入考量范围。可通过硬件或软件来执行纠正。建议步骤如下。

考虑以下 40G 和 50G 案例。对于这两者，过程是相同的，但 PMD 通道数量分别为 4 和 2。

图 38：40G 填充级别纠正示例

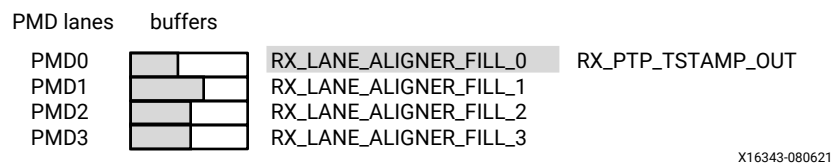


图 39：50G 填充级别纠正示例

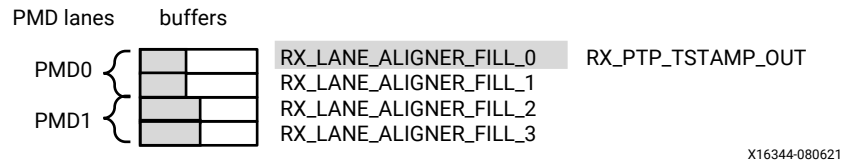


图 40：40G 偏差纠正示例

Example: RX_PTP_PCSLANE_OUT = 2

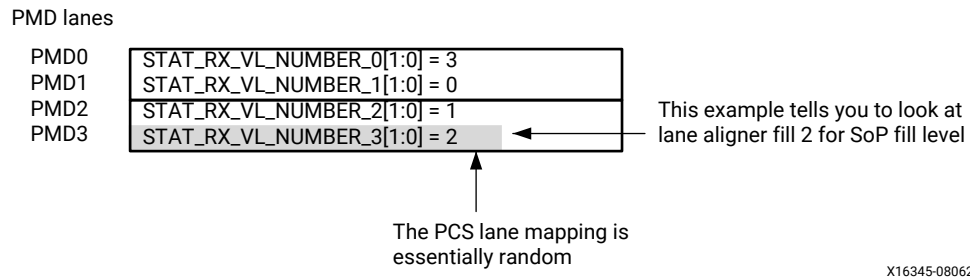


图 41：50G 偏差纠正示例

Example: RX_PTP_PCSLANE_OUT = 1

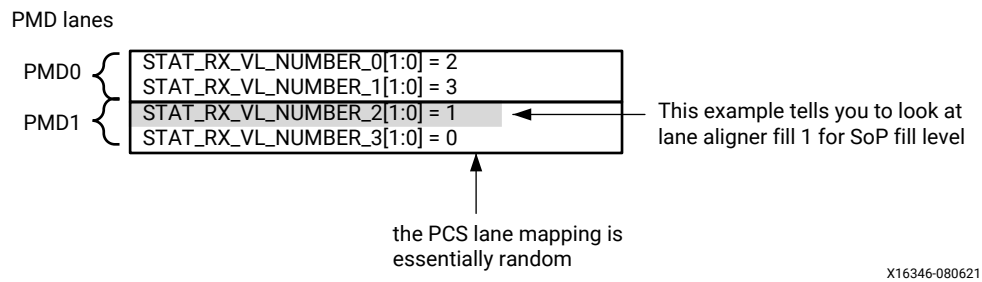
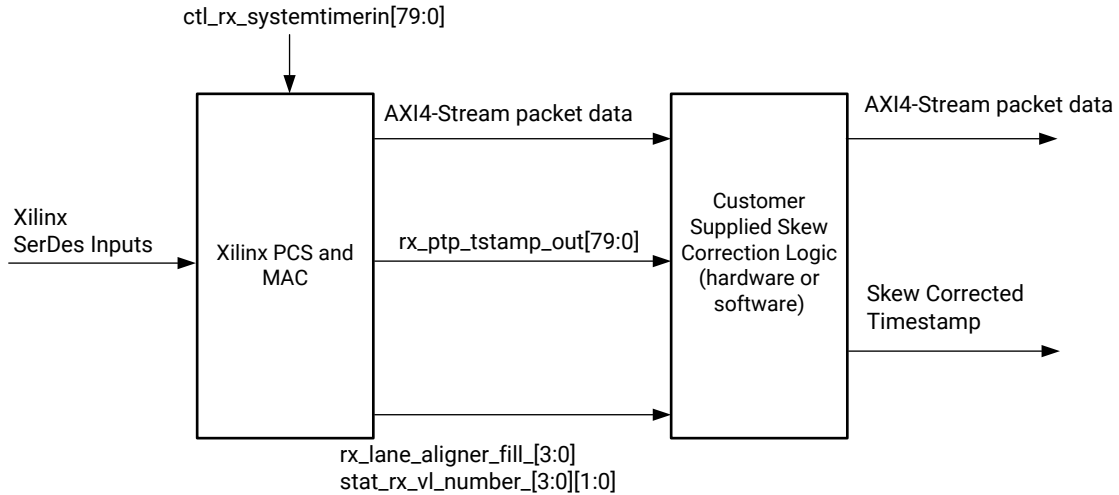


图 42：时间戳偏差纠正逻辑



X16347-080621

第一步是计算对齐缓冲器填充级别的平均时间，因为这些信号的粒度为 1 个 SerDes 时钟周期。虽然偏差在一段时间（例如，几分钟或几小时）内保持相对不变，但由于采样量化，对齐缓冲器级别的 SerDes 时钟周期存在短期浮动。因此，通过计算每个填充级别的平均时间可以获得更高精度（例如，亚纳秒级）的实际偏差。

假定已按上述方式准确判定填充级别，可使用以下公式来执行偏差纠正：

```
correction = ((RX_LANE_ALIGNER_FILL_n) - (RX_LANE_ALIGNER_FILL_0)) * SerDes
clock period
corrected timestamp = RX_PTP_TSTAMP_OUT + correction
```

其中：

corrected timestamp 即完成偏差纠正后的时间戳，必须与对应包数据保持同步

RX_PTP_TSTAMP_OUT 是捕获的时间戳。

RX_LANE_ALIGNER_FILL_0 是在其中采集时间戳的通道的对齐缓冲器填充级别，通常此通道为通道 0（请与赛灵思技术支持人员确认以了解更新信息）。

RX_LANE_ALIGNER_FILL_n 是包含 SOF 的通道的对齐缓冲器填充级别。

所有计算的单位需保持一致。由于填充级别是以时钟周期为单位来提供的，可能需将其转换为纳秒或与计算过程一致的任何单位。

如需了解其它信息，请参阅 IEEE 1588-2008 标准《IEEE 网络测量和控制系统的精密时钟同步协议标准》<http://standards.ieee.org/findstds/standard/1588-2008.html>。

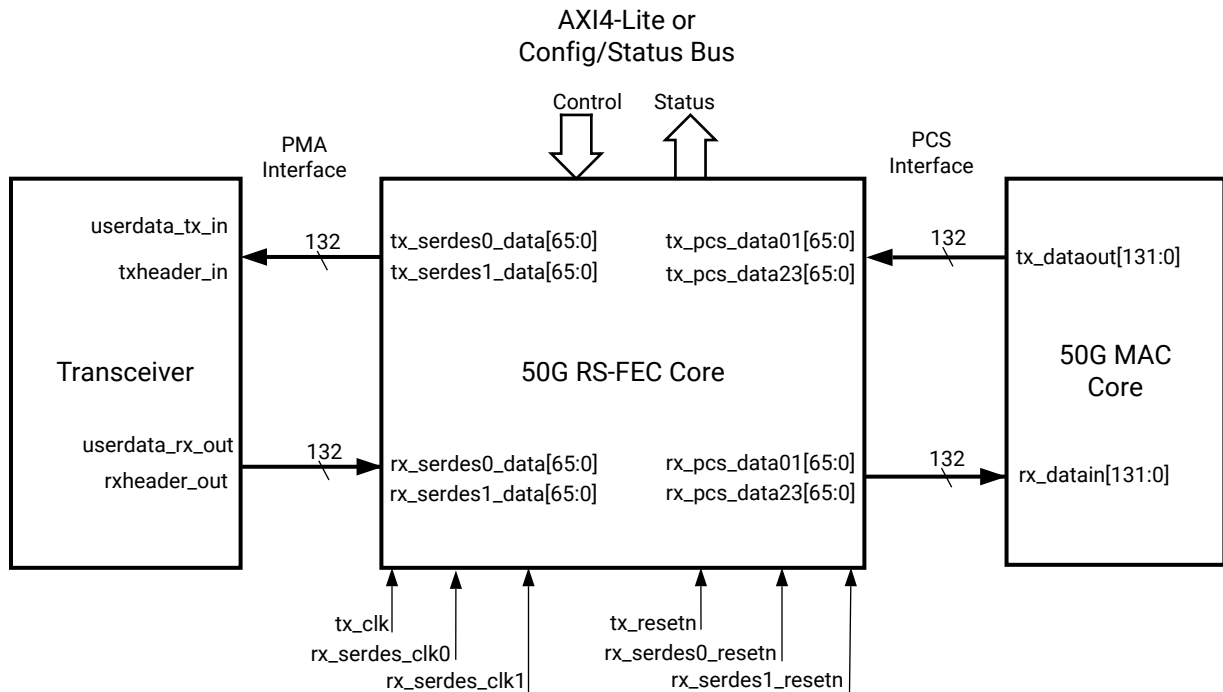
RS-FEC 支持

本节描述了 50G Ethernet 子系统的可选 RS-FEC 功能。RS-FEC 选项必须在从 IP 目录生成子系统时或者对 IP 核进行异步排序时指定。

RS-FEC 块位于 PCS 和 PMA 之间，如 [含 PCS/PMA 时钟设置的 40G/50G MAC](#) 中所示。

请参阅下图，RS-FEC 核时钟和复位与收发器信号等效，收发器复位处于高电平有效状态。

图 43：RFC 块位置



X17969-080621

RS-FEC 功能描述

50G 子系统 RS-FEC 功能可根据 50G FEC（基于 IEEE 802.3 第 91 条的以太网联盟 Schedule 3 规范）提供纠错功能。

该功能需按 IEEE 802.3 表 82-3 中的定义，插入 PCS 对齐标记。PCS 对齐标记间的字数为 20,480。

可通过使能信号绕过 RS-FEC 功能。这将绕过 RS-FEC 功能并将 PCS 直接连接到收发器，好处是可降低时延。请参阅《50G IEEE 802.3 Reed-Solomon Forward Error Correction LogiCORE IP 产品指南》(PG234)（需注册），以了解各种旁路模式下的最新时延性能数据。

有以下功能旁路模式可供选择。

- FEC Bypass Correction：解码器执行错误检测，但不纠错（请参阅 IEEE 802.3 标准第 91.5.3.3 条 [Ref 1]）。在此模式下可降低时延。
- FEC Bypass Indication：在此模式下，执行数据纠错，但无错误指示。在此模式下会生成额外信号 rx_hi_ser 用于降低无法检测到包中错误的可能性。RS 解码器会计算在代码字的非重叠块中检测到的符号错误的数量（请参阅 IEEE 802.3 标准第 91.5.3.3 条 [Ref 1]）。在此模式下可降低时延。
- Decoder Bypass：当纠错旁路和错误指示处于旁路模式时，可通过将“IEEE Error indication”设置为“Low”来绕过 RS 解码器。

RS-FEC 配置和状态端口列表及描述

下表描述了包含 RS-FEC 功能时的配置和状态端口。

表 277: RS-FEC 配置和状态端口列表及描述

名称	I/O	描述	时钟域
ctl_rsfc_enable	输入	复位后，更改才会生效。新值采样在复位后的第一个周期进行。 启用 RS-FEC 功能。1= 启用 RS-FEC 0= 绕过 RS-FEC	rx_serdes_clk
ctl_rx_rsfc_enable_correction	输入	复位后，更改才会生效。新值采样在复位后的第一个周期进行。 等效于 MDIO 寄存器 1.200.0 · 0: 解码器执行错误检测，但不执行纠错（请参阅 IEEE 802.3802.3 的第 91.5.3.3 条）。 · 1: 解码器还会执行纠错。	rx_serdes_clk
ctl_rx_rsfc_enable_indication	输入	复位后，更改才会生效。新值采样在复位后的第一个周期进行。 等效于 MDIO 寄存器 1.200.1 · 0: 绕过错误指示功能（请参阅 IEEE 802.3 标准第 91.5.3.3 条）。 · 1: 解码器向 PCS 子层发出错误指示。	rx_serdes_clk
ctl_rsfc_ieee_error_indication_mode	输入	复位后，更改才会生效。新值采样在复位后的第一个周期进行。 · 1: 核遵循 IEEE RS-FEC 规范。 · 0: 如果 ctl_rx.rsfc_enable_correction 和 ctl_rx.rsfc_enable_indication 为 0，则绕过 RS 解码器。	rx_serdes_clk
stat_tx_rsfc_pcs_block_lock	输出	TX PCS 块锁定状态 0=已解锁 1=已锁定	tx_clk
stat_tx_rsfc_lane_alignment_status	输出	TX PCS 帧对齐状态 0=未对齐 1=已对齐	tx_clk
stat_rx_rsfc_am_lock0	输出	RX 通道 1 锁定状态 0=已解锁 1=已锁定	rx_serdes_clk
stat_rx_rsfc_am_lock1	输出	RX 通道 1 锁定状态 0=已解锁 1=已锁定	rx_serdes_clk
stat_rx_rsfc_lane_alignment_status	输出	RX 对齐状态 0=未对齐 1=已对齐	rx_serdes_clk
stat_rx_rsfc_lane_fill_0[13:0]	输出	RX 通道 0 额外延迟。 每条延迟总线的头 7 个位 [13:7] 即为因去歪斜而导致添加的额外时钟周期延迟数。每条延迟总线的末 7 个位 [6:0] 均为因去歪斜而导致添加的少量时钟周期延迟，单位为时钟周期的 1/66。	rx_serdes_clk
stat_rx_rsfc_lane_fill_1[13:0]	输出	RX 通道 1 额外延迟。 每条延迟总线的头 7 个位 [13:7] 即为因去歪斜而导致添加的额外时钟周期延迟数。每条延迟总线的末 7 个位 [6:0] 均为因去歪斜而导致添加的少量时钟周期延迟，单位为时钟周期的 1/66。	rx_serdes_clk
stat_rx_rsfc_lane_mapping [1:0]	输出	RX 通道映射 位 0= PMA 通道 0 上承载的 FEC 通道的索引 位 1= PMA 通道 1 上承载的 FEC 通道的索引	rx_serdes_clk
stat_rx_rsfc_hi_ser	输出	仅当该核处于旁路指示模式下时，此输出才有效。它表示处于高电平有效时，含 8192 个代码字的窗口内 FEC 符号错误数量超过阈值 K (417)。	rx_serdes_clk
stat_rx_rsfc_corrected_cw_inc	输出	已纠正的代码字计数增量。	rx_serdes_clk
stat_rx_rsfc_uncorrected_cw_inc	输出	未纠正的代码字计数增量。	rx_serdes_clk

表 277: RS-FEC 配置和状态端口列表及描述 (续)

名称	I/O	描述	时钟域
stat_rx_rsfec_symbol_error_count0_inc[2:0]	输出	通道 0 的符号错误计数增量。	rx_serdes_clk
stat_rx_rsfec_symbol_error_count1_inc[2:0]	输出	通道 1 的符号错误计数增量。	rx_serdes_clk

状态/控制接口

状态/控制接口支持您设置 40G/50G High Speed Ethernet 子系统配置并监控 40G/50G High Speed Ethernet 子系统状态。以下各小节对部分状态信号和控制信号进行了详细描述。

RX 和 TX PCS 通道标记值

IEEE 802.3 标准定义了 PCS 通道标记值，如下表所示。BIP 字段是根据 IEEE 802.3 第 82.2.8 条计算所得的。

表 278: 40G/50G 标记定义

输入信号名称	值
ctl_rx_vl_marker_id[0][63:0] ctl_tx_vl_marker_id[0][63:0]	64'h90_76_47_BIP3_6F_89_B8_BIP7
ctl_rx_vl_marker_id[1][63:0] ctl_tx_vl_marker_id[1][63:0]	64'hF0_C4_E6_BIP3_0F_3B_19_BIP7
ctl_rx_vl_marker_id[2][63:0] ctl_tx_vl_marker_id[2][63:0]	64'hC5_65_9B_BIP3_3A_9A_64_BIP7
ctl_rx_vl_marker_id[3][63:0] ctl_tx_vl_marker_id[3][63:0]	64'hA2_79_3D_BIP3_5D_86_C2_BIP7

RX PCS 通道对齐状态

40G/50G High Speed Ethernet 子系统可提供状态位以指示字边界同步和 PCS 通道对齐的状态。所有信号均与 `clk` 的上升沿同步，以下是每个信号的详细描述。

stat_rx_synced[3:0]

当该总线的某个位为 0 时，表示对应通道的字边界同步未完成，或者发生了错误（由另一个状态位来标识）。

当该总线的某个位为 1 时，表示对应通道的字边界同步已完成，并且正在接收期望的 PCS 通道标记字。

stat_rx_synced_err[3:0]

当该总线的某个位为 1 时，表示对应通道上发生故障，可能发生的故障如下：

- 无法使用成帧位 [65:64] 在通道中完成字边界同步
- 通道中达成字边界同步后，在成帧位 [65:64] 上检测到错误
- 通道中达成字边界同步后，从未收到有效的 PCS 通道标记字

总线的位保持处于断言有效状态直至执行字边界同步为止或者直至对应通道发出某些其它错误/故障信号为止。

stat_rx_mf_len_err[3:0]

当该总线的某个位为 1 时，表示正在接收 PCS 通道标记字，但接收速度与对应通道预期不符。发射器和接收器必须以相同“Meta Frame”（元帧）长度重新配置。

总线的位保持处于断言有效状态直至执行字边界同步为止或者直至对应通道发出某些其它错误/故障信号为止。

stat_rx_mf_repeat_err[3:0]

在某个通道内达成字边界同步后，如果该总线的某个位为 1，则表示在对应通道内已检测到 4 个连续无效的 PCS 通道标记字。

总线的位保持处于断言有效状态直至执行重新同步为止，或者直至对应通道发出某些其它错误/故障信号为止。

stat_rx_mf_err[3:0]

当 `stat_rx_aligned` 值为 1 时，表示全部通道都已对齐/去歪斜，并且接收器已准备好接收包数据。

stat_rx_aligned_err

当 `stat_rx_aligned_err` 值为 1 时，将发生以下 2 种情况之一：

- 通道对齐经多次尝试后失败。
- 通道对齐已丢失（`stat_rx_aligned` 断言有效，随后被取反）。

stat_rx_misaligned

当 `stat_rx_misaligned` 值为 1 时，表示未能在所有 PCS 通道上同时接收到有效的 PCS 通道标记字。

每次检测到此错误状况时，就会断言此输出有效并保持 1 个时钟周期。

stat_rx_framing_err [3:0][3:0] 和 stat_rx_framing_err_valid [3:0]

这组总线旨在用于保留同步报头错误记录。每个 PCS 通道都有一对输出。

`stat_rx_framing_err_[PCSL_LANES-3:0]` 输出总线用于指示已接收到的同步报头错误数量，当对应 `stat_rx_framing_err_valid_[PCSL_LANES-3:0]` 采样为 1 时，此输出总线合格（即，仅限在此情况下其值才有效）。

stat_rx_vl_number[3:0][1:0]

每个总线所指示的特定 PCS 通道的状态反映在特定状态管脚上。例如，`stat_rx_vlane_number_0` 所指示的 PCS 通道的状态反映在其它状态信号的管脚 0 上。

这些总线可用于检测是否尚未找到 PCS 通道或者是否某一 PCS 通道已映射到多个状态管脚。

stat_rx_vl_demuxed[3:0]

在每个通道上都达成字边界同步后，如果该总线的位为 1，则表示已正确查找对应 PCS 通道，并且已对其正确完成逆多路复用。

stat_rx_block_lock[3:0]

每个位都用于指示对应 PCS 通道已达成 IEEE 802.3-2015 标准所定义的同步报头锁定。值为 1 表示已达成块锁定。

stat_rx_status

当 `stat_rx_aligned` 为 1 且 `stat_rx_hiber` 为 0 时，此输出设置为 1。此设置由 IEEE 802.3-2015 标准定义。

stat_rx_local_fault

当 `stat_rx_received_local_fault` 或 `stat_rx_internal_local_fault` 断言有效时，此输出设置为 1。此输出对电平敏感。

RX 错误状态

40G/50G High Speed Ethernet 子系统提供了一些状态信号，用于指示 64 位/66 位代码字和序列违例以及 CRC32 校验故障。

所有信号都与 `clk` 上升沿同步，下面提供了每个信号的详细描述。

stat_rx_bad_fcs

当此信号值为 1 时，表示错误检测逻辑已在接收到的包中发现接收的 CRC32 值与期望的值不匹配。

当检测到 CRC32 错误时，接收到的包将被标记为包含错误，并随在上一次传输（即 `rx_eopout` 断言有效的周期）期间断言有效的 `rx_errout` 一起发送，但 `ctl_rx_ignore_fcs` 断言有效的情况下除外。

每次检测到 CRC32 错误时，都将断言此信号有效并保持 1 个时钟周期。

stat_rx_bad_code[1:0]

此信号用于指示 RX PCS 接收状态机处于 IEEE 802.3-2015 标准所定义的 `RX_E` 状态的周期数。

设计流程步骤

本章节描述了子系统的自定义和生成方式、子系统的约束方式以及此 IP 子系统的仿真、综合与实现的具体步骤。如需获取有关标准 Vivado® 设计流程以及有关 IP integrator 的详细信息，请参阅以下 Vivado Design Suite 用户指南：

- 《Vivado Design Suite 用户指南：采用 IP integrator 设计 IP 子系统》(UG994)
- 《Vivado Design Suite 用户指南：采用 IP 进行设计》(UG896)
- 《Vivado Design Suite 用户指南：入门指南》(UG910)
- 《Vivado Design Suite 用户指南：逻辑仿真》(UG900)

自定义和生成子系统

本节包含有关如何使用赛灵思工具在 Vivado® Design Suite 中自定义和生成子系统的信息。

如果要在 Vivado IP integrator 中自定义和生成子系统，请参阅《Vivado Design Suite 用户指南：采用 IP integrator 设计 IP 子系统》(UG994) 了解详情。确认或生成设计时，IP integrator 可能会自动计算某些配置值。要查看配置值是否会更改，请参阅本章中的参数说明。要查看参数值，请在 Tcl 控制台中运行 `validate_bd_design` 命令。

您可以遵循以下步骤通过指定与 IP 子系统关联的各种参数值来自定义设计中使用的 IP：

1. 从 IP 目录选择 IP。
2. 双击所选 IP，或者从工具栏或右键单击菜单中选择“Customize IP”（自定义 IP）命令。

欲知详情，请参阅《Vivado Design Suite 用户指南：采用 IP 进行设计》(UG896) 和《Vivado Design Suite 用户指南：入门指南》(UG910)。

本章中的附图是 Vivado IDE 的插图。此处展示的布局可能与当前版本中的布局有所不同。

适用于 Versal ACAP 的赛灵思 IP - GT 四通道集成

基于赛灵思 GT 的 IP（如 Aurora、PCIe®、L Ethernet 和 MRMAC）可在 IP integrator 中提供块自动化设置，以支持您将多个父级赛灵思 IP 无缝连接到 GT 四通道。IP 块自动化设置可例化 GT 四通道，并创建基本数据路径 `USRCLK` 和 `GT REFCLK` 连接。

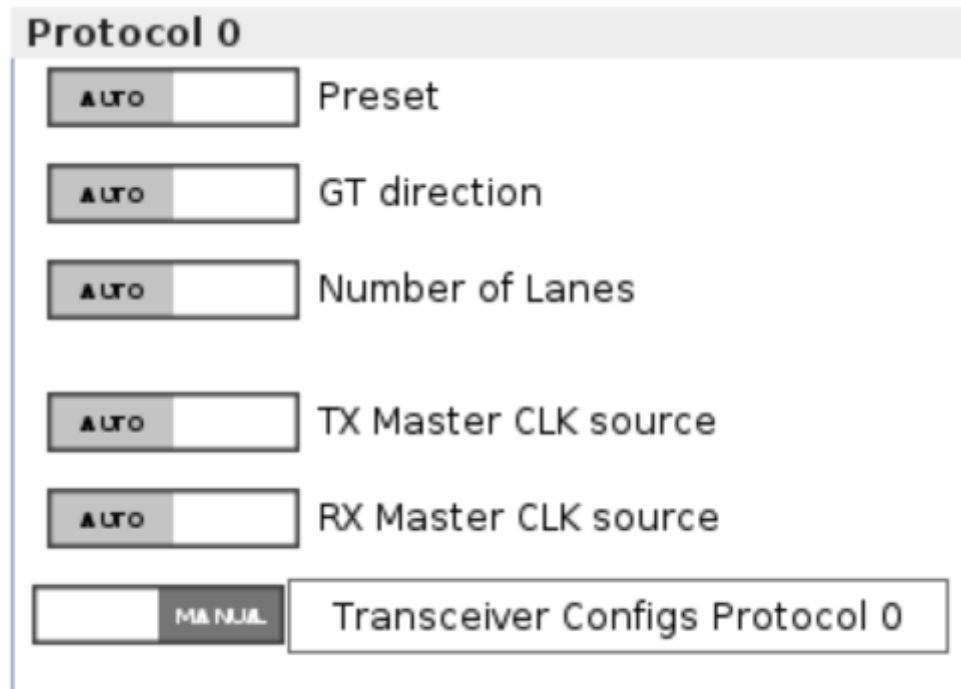
要使用块自动化设置来连接多个 L Ethernet IP，请执行以下步骤：

1. 使用 IP integrator 画布中的“Add IP”（添加 IP）选项添加“L Ethernet”IP。
2. 配置“L Ethernet”IP 的通道数量、线速率等。
3. 单击“Run_Block_Automation”。在“Block Automation”（块自动化设置）GUI 中，选择下列选项之一：“Auto”（自动）、“Start_with_New_Quad”（以新四通道开始）或“Customized_Connections”（自定义连接）。

4. 根据您的系统需求，执行步骤 2 和 3 以添加更多“L Ethernet” IP 实例。

确认设计时，会将 GT 四通道参数从相连的 IP 传输至这些 GT 四通道。这样即可在“Transceiver Wizard” GUI 中将所有 GT 四通道参数标记为“Auto”（自动）。但您可在“Transceiver Configs”（收发器配置）中将“Auto”选项更改为“Manual”（手动），以便对“insertion loss”（插入损耗）、“drive strength”（驱动强度）、“equalization”（均衡）等参数以及其它高级设置进行微调，如下图所示。切换为“Manual”模式后，无论对父级 IP 配置进行任何更改，而后执行确认，都不会再将任何 GT 四通道参数从父级 IP 传输至 GT 四通道。因此，仅限在将所有必要的父级 IP 参数都传输到 GT 四通道后，才能执行手动更改。

图 44：“Transceiver Wizard” 中的“Auto”到“Manual”选项切换



“Configuration” 选项卡

“Configuration”（配置）选项卡提供了基本核配置选项。

所有选项卡中的默认值均已预填充。

图 45：适用于 UltraScale/UltraScale+ 的“Configuration”选项卡

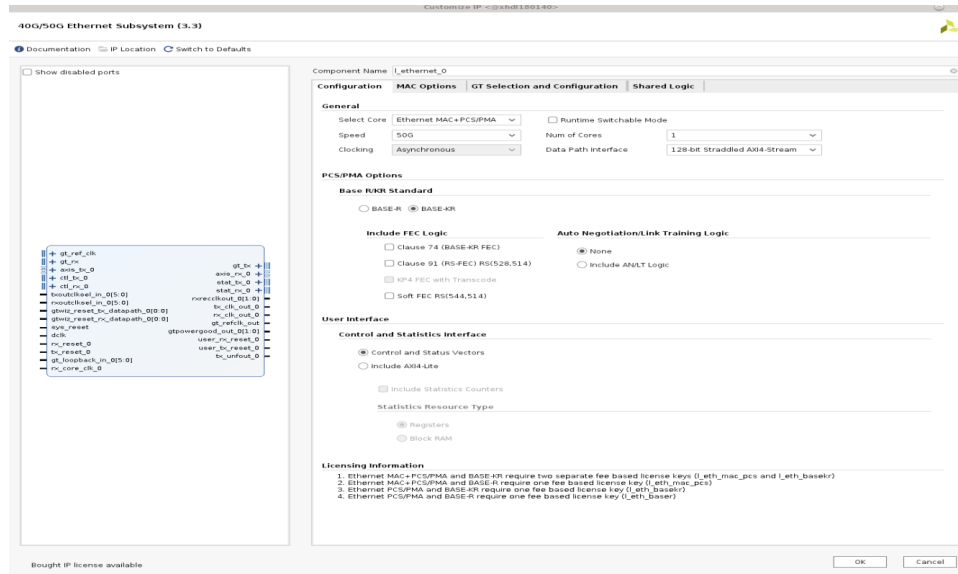


图 46：适用于 Versal ACAP 的“Configuration”选项卡

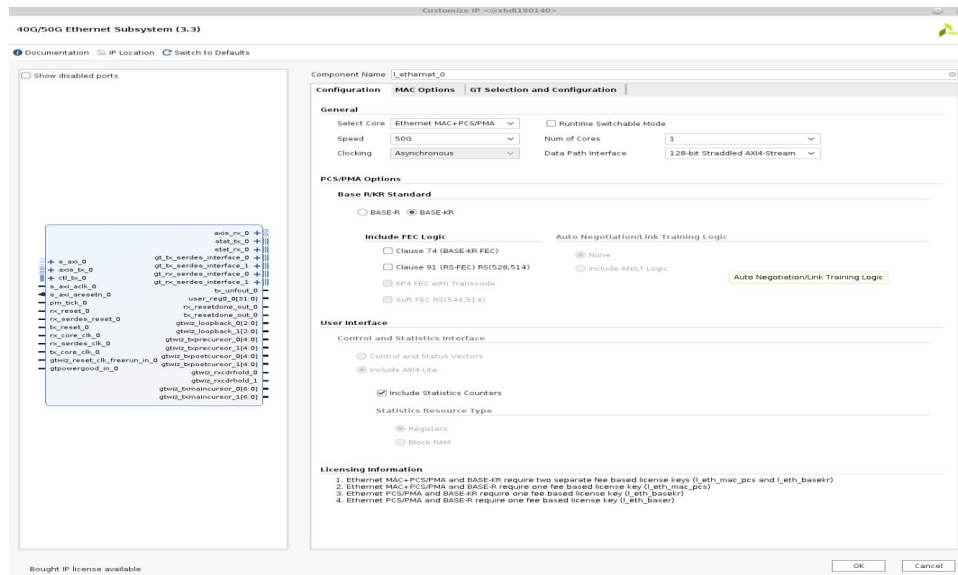


表 279：配置选项

选项	值	默认
General		
Select Core	Ethernet MAC+PCS/PMA Ethernet PCS/PMA	Ethernet MAC+PCS/PMA
Speed	50G 40G	50G
Number of Cores	1 2	1

表 279：配置选项 (续)

选项	值	默认
Clocking	Synchronous Asynchronous	Asynchronous
Runtime Switchable Mode	0 和 1	0
Data Path Interface	128-bit Straddled AXI4-Stream ¹ 256-bit Regular AXI4-Stream ² MII ³	128-bit Straddled AXI4-Stream
PCS/PMA Options		
Base-R Base-KR	Base-R Base-KR	Base-KR
Include FEC Logic		
Clause 74 (BASE-KR FEC) ^{4,6}	0 和 12 mi	0
Clause 91 (RS-FEC) ^{5,6}	0 和 1	0
KP4 FEC with Transcode ⁷	0 和 1	0
Soft RS-FEC (544,514)	0 和 1	0
Auto Negotiation/Link Training Logic		
Auto Negotiation/Link Training Logic	None Include AN/LT Logic	None
Control and Statistics Interface		
Control and Statistics interface	Control and Status Vectors Include AXI4-Lite	Control and Status Vectors
Include Statistics Counters	0 和 1 ⁸	1
Statistics Resource Type	Register 和 block RAM ⁹	block RAM

注释：

- “128-bit Straddled AXI4-Stream” 是对含 40G 和 50G 线速率选项的以太网 MAC+PCS/PMA 可见的选项。
- “256-bit Regular AXI4-Stream” 是对仅含 40G 线速率的以太网 MAC+PCS/PMA 可见的选项。
- MII 接口是对以太网 PCS/PMA 核可见的唯一选项。
- 针对 Base-R 不支持 “Clause 74 (BASE-KR FEC)” 逻辑。
- 针对 40G 速度的 Base-R 不支持 “Clause 91 (RS-FEC)”。
- “Clause 74 (BASE-KR FEC)” 和 “Clause 91 (RS-FEC)” 在 Vivado® IDE 中均为可选，但在正常操作期间每次只能使用相应的控制信号启用其中之一。
- GTM 集成硬核 FEC。仅适用于含 GTM 的器件（不适用于 Versal® ACAP）。
- 仅当启用 “Include Statistics Counters”（包含统计数据计数器）选项时，在寄存器映射中 “Statistics Counters”（统计数据计数器）才可用。否则，“Statistics Counters” 不可用。
- “Statistics Resource Type block RAM”（统计数据资源类型块 RAM）选项将在后续版本中提供。
- 针对 Versal ACAP 仅提供单核支持。由于设计示例的 GT 位于核外部，因此不支持多核。

“MAC Options” 选项卡

“MAC Options” 选项卡提供额外的核配置选项。

图 47：适用于 UltraScale/UltraScale+ 的“MAC Options”选项卡

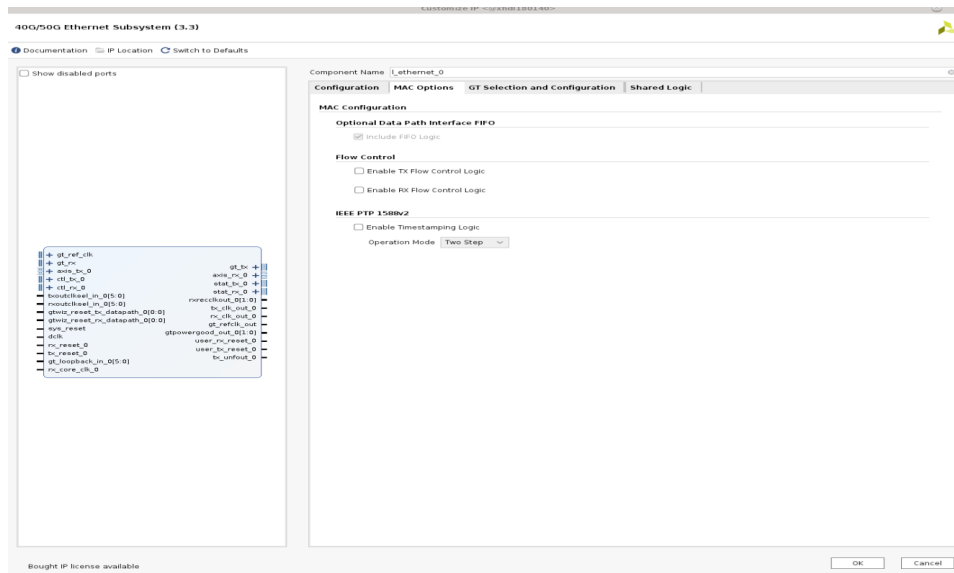


图 48：适用于 Versal ACAP 的“MAC Options”选项卡

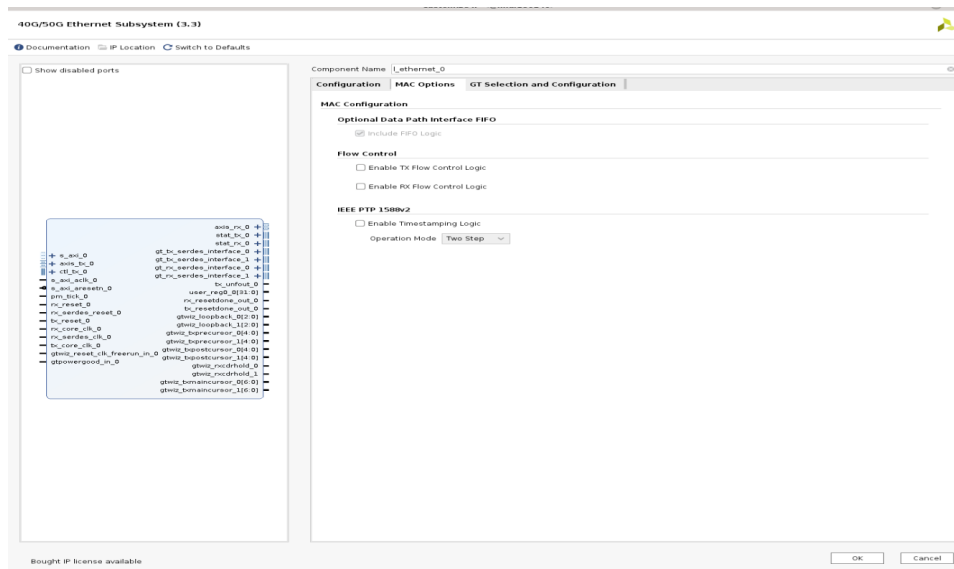


表 280：MAC 配置

选项	值	默认
可选数据路径接口 FIFO		
Include FIFO Logic	0 和 1	1
流量控制		
Enable TX Flow Control Logic	0 和 1	0
Enable RX Flow Control Logic	0 和 1	0

表 280: MAC 配置 (续)

选项	值	默认
IEEE PTP 1588v2		
Enable Timestamping Logic	0 和 1	0
Operation Mode	Two Step	Two Step

“GT Selection and Configuration” 选项卡

“GT Selection and Configuration” 选项卡支持您配置核的串行收发器功能。

图 49: 适用于 UltraScale/UltraScale+ 的 “GT Selection and Configuration” 选项卡

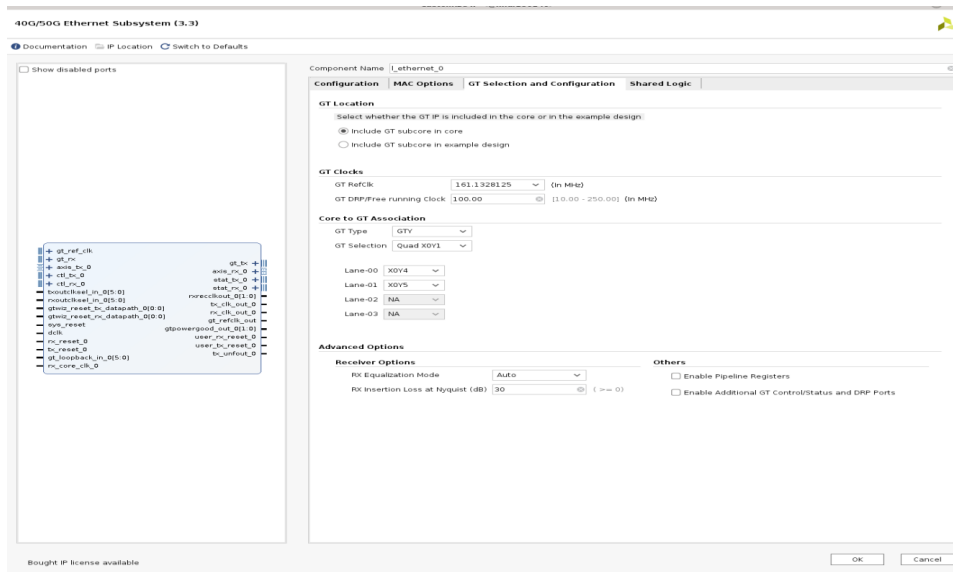


图 50：适用于 Versal ACAP 的“GT Selection and Configuration”选项卡

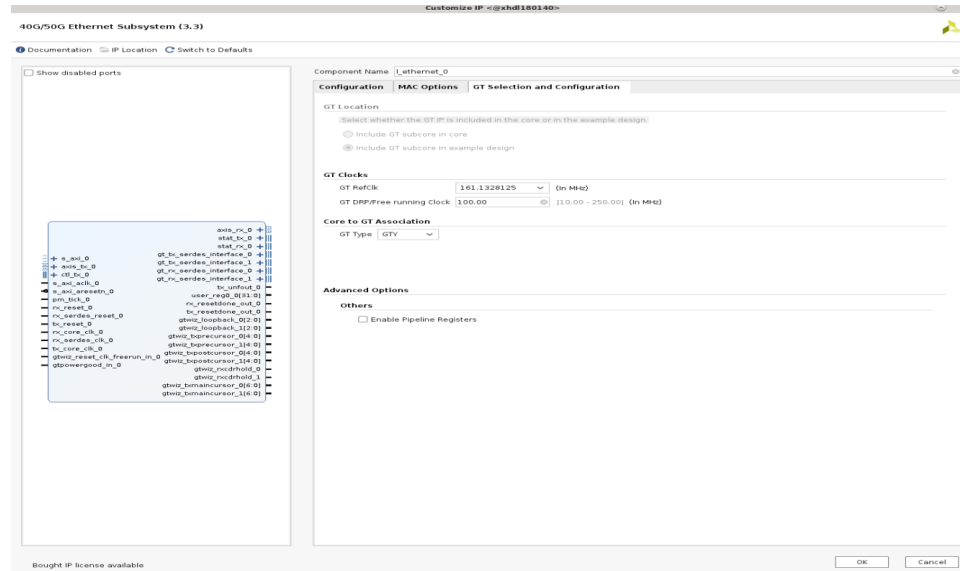


表 281：GT 时钟选项

选项	值	默认
GT Location		
选择将 GT IP 包含在核中还是包含在设计示例中	<input type="radio"/> Include GT subcore in core <input checked="" type="radio"/> Include GT subcore in example design	Include GT subcore in core
GT Clock³		
GT RefClk (MHz) ¹	161.1328125	161.1328125
	195.3125	
	201.4160156	
	257.8125	
	322.265625	
	312.5	
GT DRP Clock (MHz)	10.00 - 250.00 MHz	100.00
Core to Transceiver Association		
GT Type	GTH、GTY、GTM 和 GTYP	GTY
GT Selection	选项基于器件/封装四通道组。例如： Quad X0Y1 Quad X0Y2 Quad X0Y3 GTM_DUAL_X0Y0 如果 GT 类型选为 GTM，则 GTM 双通道列表可供选择。例如：GTM_DUAL_X0Y0 GTM_DUAL_X0Y1	Quad X0Y0

表 281：GT 时钟选项 (续)

选项	值	默认
Lane-00 到 Lane-03	根据器件/封装自动填充。 例如，如果 Speed = 50G 且 Num of Cores = 2（或者 Speed = 40G 且 Num of Cores = 1）并且 GT Selection = Quad X0Y1，则 4 个 GT 通道为： X0Y4 X0Y5 X0Y6 X0Y7	
RX Equalization Mode	Auto LPM DFE	Auto
RX Insertion Loss at Nyquist (dB)	取决于 GT Wizard	30
其它		
Enable Pipeline Register	0 和 1	0
Enable Additional GT Control and Status Ports	0 和 1	0

注释：

1. 此列表提供用于默认配置的频率。如需获取不同速度所支持的时钟频率的完整列表，请参阅最新版工具中的 Vivado IDE。
2. 仅适用于 GTM。对于 40G GTM，需要选择 2 个连续双通道。
3. 支持的 GT 时钟：
 - a. 40G 支持 156.25、103.125、128.90625、156.25、161.1328125、206.25、257.8125、309.375、312.5 和 322.265625 GT RefClk 频率（以 MHz 为单位）。
 - b. 50G 支持 161.1328125、161.1328125、156.25、312.5、195.3125、201.4160156、257.8125 和 322.265625 GT RefClk 频率（以 MHz 为单位）。
 - c. 50G GTM (UltraScale+ 器件) 支持 161.1328125、156.25、161.1328125、195.3125、201.4160156、257.8125 和 322.265625 GT RefClk 频率（以 MHz 为单位）。
 - d. 运行时切换功能支持 128.90625、128.90625、206.25、257.8125、156.25、312.5、161.1328125 和 322.265625 gt_refclk 频率（对于 Versal 器件）以及 161.1328125 gt_refclk 频率（对于非 Versal 器件）（以 MHz 为单位）。

“Shared Logic” 选项卡

“Shared Logic” 选项卡可让您在核或设计示例中使用共享逻辑。

图 51：适用于 UltraScale/UltraScale+ 的“Shared Logic”选项卡

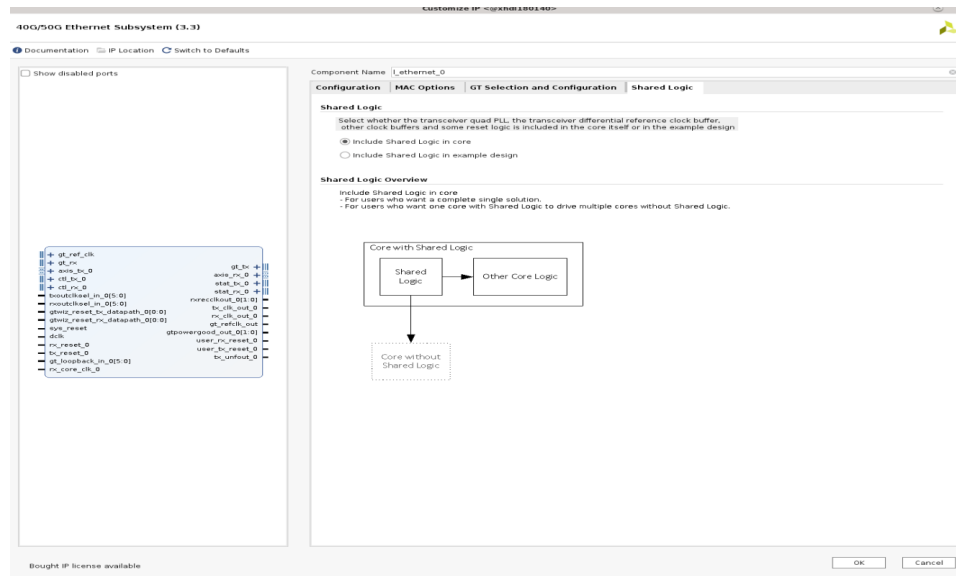


表 282：“Shared Logic”（共享逻辑）选项

选项	默认
Include Shared Logic in Core	Include Shared Logic in Core
Include Shared Logic in example design	

配置电子数据表

除了 Vivado 设计工具中的 IP 目录外，还可以利用客户提供的电子数据表请求 40G/50G Ethernet Subsystem 进行特定配置。您必须向赛灵思技术支持提交填写妥的电子数据表，才能获取目标网表。如需获取有关应提交哪些内容以及如何提交进行提交的描述，请参阅“综合与实现”。

如需全面了解核，请参阅与所选的核及器件对应的文档，例如《UltraScale 架构 GTH 收发器用户指南》(UG576)。

相关信息

[综合与实现](#)

输出生成

如需了解更多详情，请参阅《Vivado Design Suite 用户指南：采用 IP 进行设计》(UG896)。

约束子系统

每个版本都包含 1 个或多个您的配置及其时钟设置专用的 XDC 文件。将 IP 整合到设计中时，可能还需要其它约束。如需更多信息，请联系赛灵思技术支持。

所需约束

本节不适用于此 IP 子系统。

器件、封装和速度等级选择

本节不适用于此 IP 子系统。

时钟频率

本节不适用于此 IP 子系统。

时钟管理

本节不适用于此 IP 子系统。

时钟布局

本节不适用于此 IP 子系统。

bank 分配

本节不适用于此 IP 子系统。

收发器布局

本节不适用于此 IP 子系统。

I/O 标准与布局

本节不适用于此 IP 子系统。

仿真

有关 Vivado® 仿真组件的全面信息，以及与如何使用支持的第三方工具相关的信息，请参阅《Vivado Design Suite 用户指南：逻辑仿真》(UG900)。

每个版本都包含一个演示仿真测试激励文件。在包含的加密 RTL 上执行仿真。测试激励文件包括从用户接口的 TX 侧环回、通过 TX 电路、环回到 RX 电路并检查用户接口的 RX 侧接收到的包。

环回仿真包括穿过收发器的路径。仿真是使用专为多个常见的业界标准仿真器所提供的 Linux 脚本来运行的。

如需了解更多信息，请参阅 [第 7 章：测试激励文件](#)。

由于完成对齐所需时间较长，仿真完成所需时间可能较长。可通过 ``define SIM_SPEED_UP` 来缩短仿真时间，它可通过减少 PCS 通道对齐标记 (AM) 间隔来帮助 IP 加速实现对齐。设置 `define SIM_SPEED_UP` 将减少 `CTL_TX_VL_LENGTH_MINUS1` 和 `CTL_RX_VL_LENGTH_MINUS1`。

`SIM_SPEED_UP` 选项在下列情况下可用于加速仿真：在串行环回中，或者在两个端点处都可减少对齐标记间隔的情况下。该选项与使用串行环回的设计示例仿真兼容。

注释：

- SIM_SPEED_UP 仅在运行 RTL 仿真时才可用。它在使用综合后或实现后网表运行仿真时不可用。
- 更改 CTL_TX_VL_LENGTH_MINUS1 和 CTL_RX_VL_LENGTH_MINUS1 的默认值将违反 IEEE 802.3 规范。
- 减少 AM 间隔将导致链路上的可用带宽减小。使用仅限 PCS 核时，TX 核不向用户逻辑提供反压，并且减小的带宽可能在不间断连续传输大量包时导致 stat_tx_fifo_error。
- 此更改只能在仿真中执行。要使设计能在硬件中正常运行，必须使用默认 IEEE 值。
- 全速率仿真仍应在不使用 SIM_SPEED_UP 选项的情况下运行。

VCS

使用 vlogan 选项：+define+SIM_SPEED_UP。

ModelSim

使用 vlog 选项：+define+SIM_SPEED_UP。

Questa Advanced Simulator

使用 vlog 选项：+define+SIM_SPEED_UP

Xcelium Parallel Simulator

使用 xmvlog 选项：+define+SIM_SPEED_UP

Vivado 仿真器

使用 xvlog 选项：-d SIM_SPEED_UP

Aldec Riviera-PRO 仿真器

使用 vlog 选项：+define+SIM_SPEED_UP

启用 RS-FEC 的配置仿真

为加速仿真，可应用 SIM_SPEED_UP 并在“Settings”（设置）窗口中取消选中“Use Precompiled IP simulation libraries”（使用预编译的 IP 仿真库）复选框，如下图所示。如不执行此操作，仿真可能长时间运行并导致超时错误。

图 52: “Use Pre-compiled IP Simulation Libraries” 已禁用

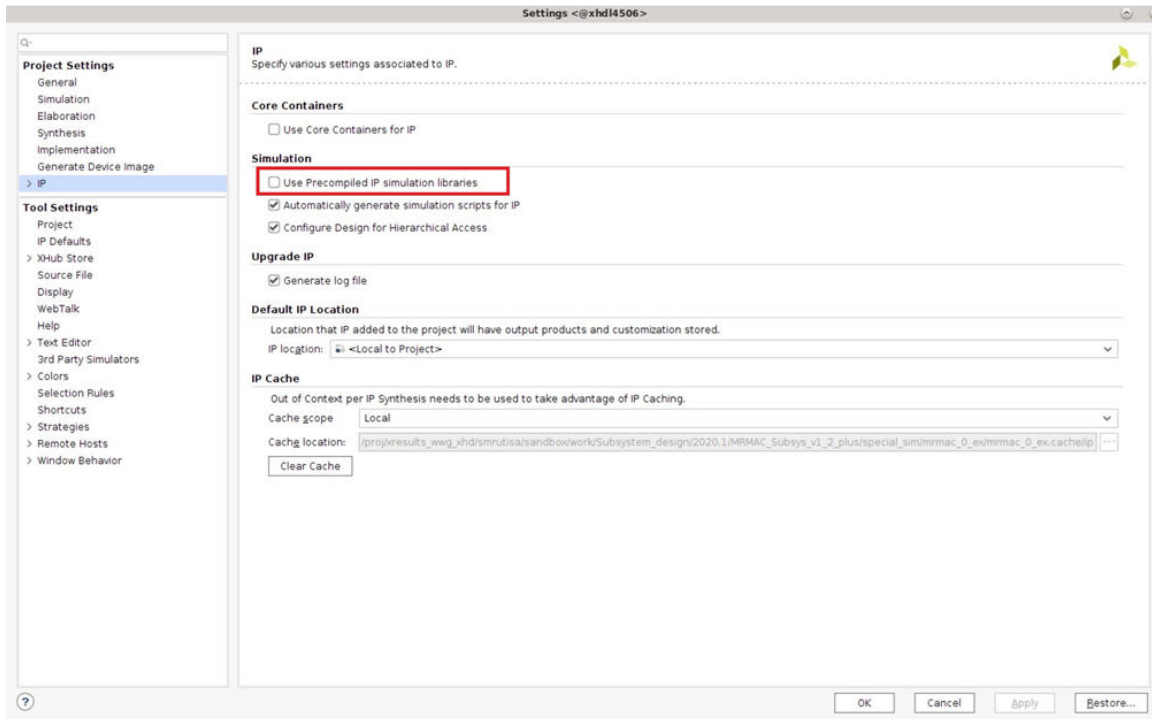
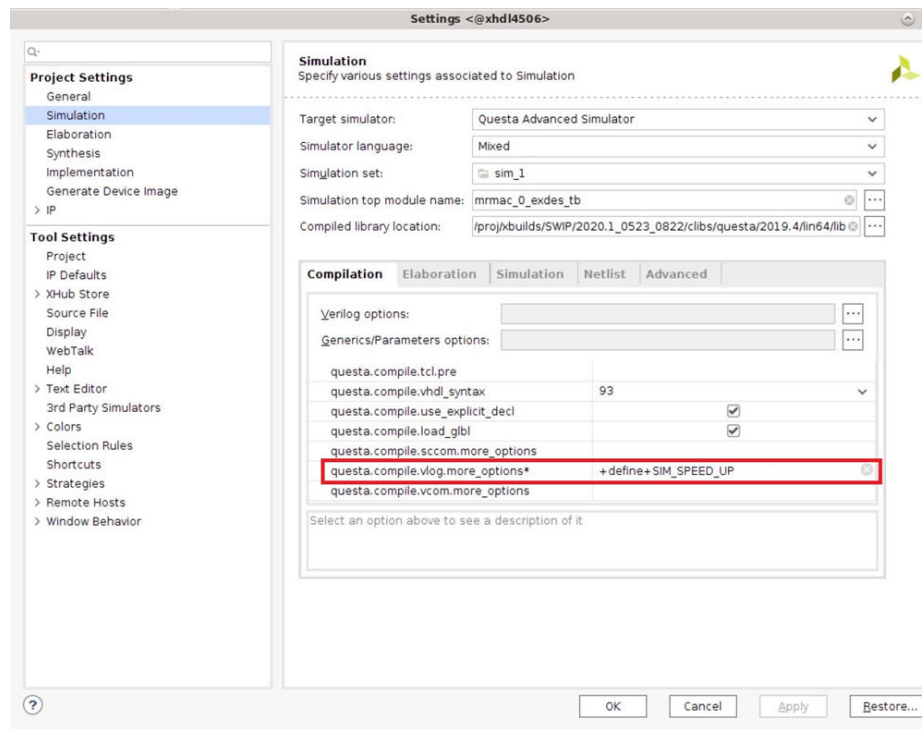


图 53: SIM_SPEED_UP 已启用



综合与实现

如需了解有关综合和实现方面的详情，请参阅《Vivado Design Suite 用户指南：采用 IP 进行设计》(UG896)。

设计示例

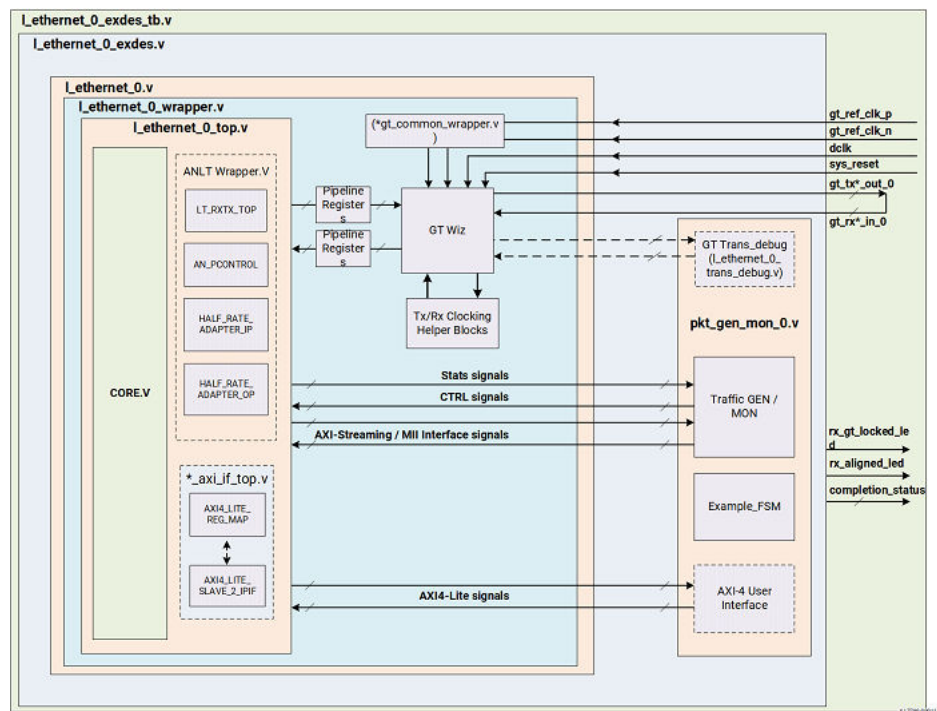
本章提供了 40G/50G Ethernet Subsystem 设计示例的简要说明。

设计示例层级

下图显示了当 GT（串行收发器）位于 UltraScale™/UltraScale+™ 器件的 IP 核内时，L_ethernet_0 单核配置设计示例的各个模块的例化及其层级。重定时寄存器用于核与 GT 之间的数据同步。时钟帮助程序块用于为该核生成所需时钟频率。

对于 Versal® 平台，gt_quad_base（适用于 Versal 的 GT Wizard）只能是设计示例的一部分。将使用 IP integrator（块自动化装置）在块设计中连接 40G/50G High Speed Ethernet IP 和 GT（串行收发器）IP。图 56：设计示例层级中含 GT 的单核 (UltraScale/UltraScale+) 显示了 Versal ACAP 的单核配置的层级。

图 54：单核设计示例层级 (UltraScale/UltraScale+)



下面是适用于不同配置的用户接口：

- MAC/PCS 配置
 - AXI4-Stream, 表示数据路径接口
 - AXI4-Lite, 用于控制接口和统计数据接口
- PCS 配置
 - MII, 表示数据路径接口
 - AXI4-Lite, 用于控制接口和统计数据接口

`_l_ethernet_0_pkt_gen_mon` 模块用于生成数据包以进行完整性测试。包生成和检查由有限状态机 (FSM) 模块来控制。

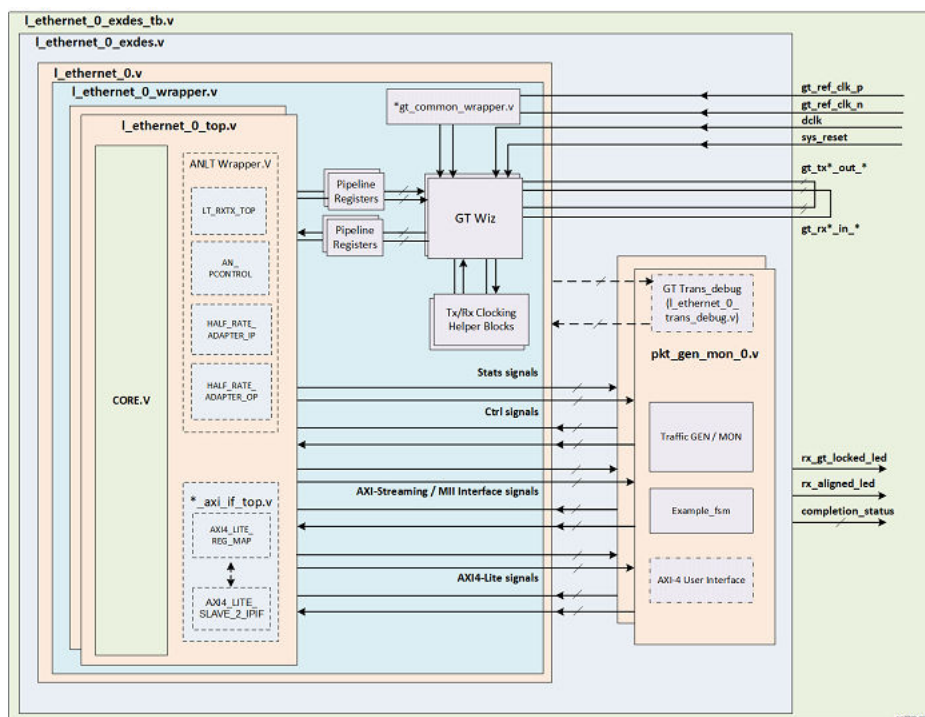
可选模块的描述如下：

- `_l_ethernet_0_trans_debug`：在下列情况下，设计示例中存在此模块：在 Vivado® 集成设计环境 (IDE) 中，启用“GT Selection and Configuration” (GT 选择和配置) 选项卡下的“Additional GT Control and Status Ports” (其它 GT 控制和状态端口) 复选框、在“GT Selection and Configuration” 选项卡下启用“Include GT subcore in example design” (在设计示例中包含 GT 子核) 选项，或者在“Configuration” (配置) 选项卡下启用“Runtime Switchable mode” (运行时可切换模式) 选项。此模块从 `_l_ethernet` 核中引出所有 GT 通道 DRP 端口以及收发器模块的部分控制端口和状态端口。
- 重定时寄存器：在“GT Selection and Configuration” 选项卡下选中“Enable Retiming Register” (启用重定时寄存器) 选项时，将在核与 GT 之间包含一个单阶流水线寄存器以简化时序，其中使用 `gt_txusrclk2` 表示 TX 路径，使用 `gt_rxusrclk2` 表示 RX 路径。但默认情况下，针对 GT 与核之间的信号完成二阶寄存。

注释：对于 Runtime Switchable，如果在 Vivado IDE 中选中“Auto Negotiation/Link training” (自动协商/链路训练)，那么在切换开关期间将仅以 40G 数据速率执行 AN 操作，并在任务模式下执行 LT 操作。

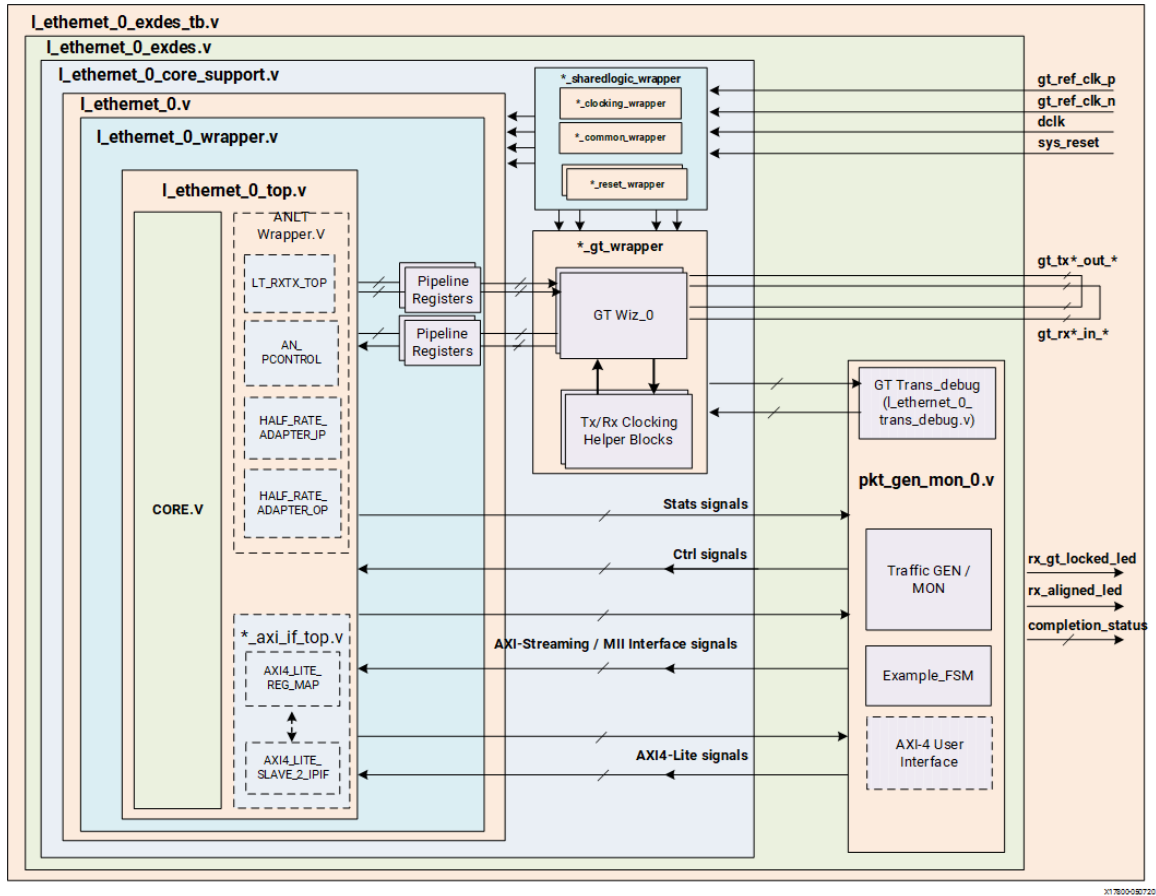
下图显示了 `_l_ethernet_0` 多核配置设计示例的各个模块的例化及其层级。

图 55：多核设计示例层级



设计示例层级 (设计示例中包含 GT)

图 56: 设计示例层级中含 GT 的单核 (UltraScale/UltraScale+)



107000-08720

下图显示了当 GT 位于设计示例中时，l_ethernet_0 设计示例的多核配置的各模块的例化及其层级。

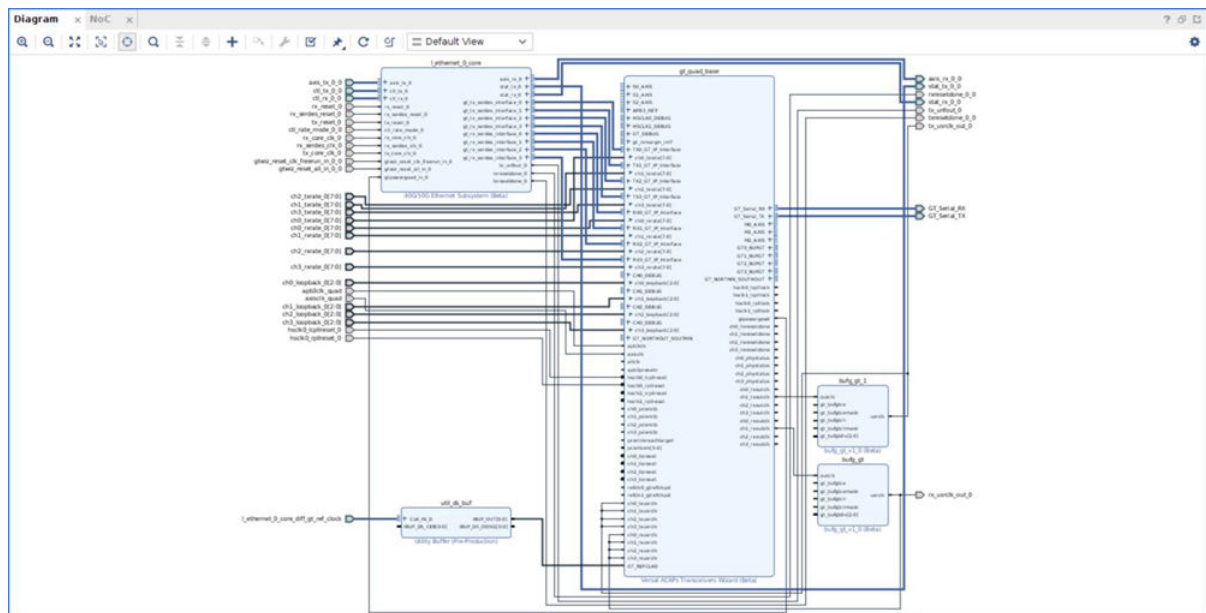
对于 Versal® 平台，gt_quad_base（对应 Versal 的 GT Wizard）将仅包含在设计示例中；40G/50G High Speed Ethernet IP 和 GT（串行收发器）IP 将在块设计中使用 IP integrator（块自动化设置）来连接。

下图显示了块设计，其中在 IP integrator 中已连接 40/50G 以太网设计示例。如需了解有关 IP integrator 的更多信息，请参阅《Vivado Design Suite 教程：采用 IP integrator 设计 IP 子系统》(UG995)。

将 40G/50G High Speed Ethernet 子系统添加到 Vivado® IP integrator 中并运行“Block Automation”（块自动化设置）时，IP/核及 GT（串行收发器）将根据核配置与部分帮助程序块相连。在 40G/50G High Speed Ethernet IP 内部有复位接口 IP，用于将 tx/rx mstreset 释放到 Versal GT、检查 tx/rx mstresetdone 状态，并将 GT 排序复位。

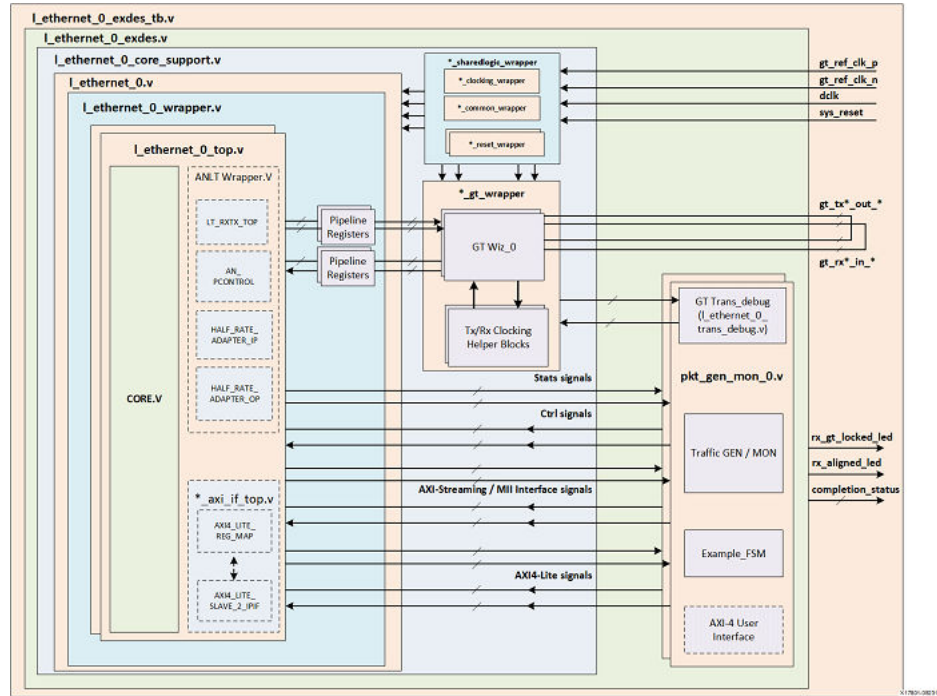
注释：只要 40G/50G High Speed Ethernet 子系统核配置中发生任何更改，就会运行确认设计并确保可通过确认。这样即可确认所有更改都已应用/传输到 IP integrator 中的 GT。

图 58：含 Versal GT 的 40/50G Ethernet Subsystem 块设计



用户接口

图 59：按设计示例层级划分的含 GT 的多核



下表中的 GPIO 用于控制设计示例。

表 283：用户输入/输出端口

名称	大小	I/O	描述
sys_reset	1	输入	用于 I_ethernet 核的复位。
gt_refclk_p	1	输入	差分输入时钟，连接到 GT。此时钟频率应与 Vivado IDE 的“GT Selection and Configuration”选项卡中所提及的 GT RefClk 频率相等。
gt_refclk_n	1	输入	差分输入时钟，连接到 GT。此时钟频率应与 Vivado IDE 的“GT Selection and Configuration”选项卡中所提及的 GT RefClk 频率相等。
dclk	1	输入	稳定/自由运行的输入时钟，连接到 GT。此时钟频率应与 Vivado IDE 的“GT Selection and Configuration”选项卡中所提及的 GT DRP 时钟频率相等。
rx_gt_locked_led	1	输出	指示 GT 处于已锁定状态。
rx_aligned_led	1	输出	指示已完成 RX 对齐。

表 283：用户输入/输出端口 (续)

名称	大小	I/O	描述
completion_status			<p>此信号表示测试状态/结果。</p> <ul style="list-style-type: none"> · 5'd0: 测试未运行。 · 5'd1: 50GE/40GE 核测试已成功完成，并已通过 · 5'd2: 没有任何通道上存在块锁定。 · 5'd3: 并非所有通道都已实现块锁定。 · 5'd4: 部分通道实现块锁定后丢失块锁定。 · 5'd5: 没有任何通道上存在通道同步。 · 5'd6: 并非所有通道都已实现同步。 · 5'd7: 部分通道实现同步后丢失同步。 · 5'd8: 未实现任何对齐状态或 rx_status。 · 5'd9: 实现对齐状态和 rx_status 后，其中之一丢失。 · 5'd10: TX 超时。 · 5'd11: 未发送任何 tx 数据。 · 5'd12: 接收到的包数量与发送的包数量不相等。 · 5'd13: 接收到的字节总数与发送的字节总数不相等。 · 5'd14: 检测到协议错误。 · 5'd15: 在接收到的包中检测到位元错误。 · 5'd31: 测试滞留在复位状态。
restart_tx_rx_*	1	输入	当包生成器和包监控器处于空闲状态时，此信号用于重新启动包生成和接收以便执行数据完整性测试。
send_continuous_pkts_*	1	输入	<p>此端口可用于发送连续包以供开发板确认。</p> <ul style="list-style-type: none"> · 1'b0: 发送固定的 20 个包以供仿真。 · 1'b1: 发送连续包以供开发板验证。
mode_change_0	1	输入	仅当在 Vivado IDE 中选中“Runtime Switchable”（运行时可切换）且用于切换核速度时，此端口才可用。
core_speed_0	1	输出	此信号用于指示核的工作速度：1'b1 = 40G，1'b0 = 50G。

核 xci 顶层端口列表

在下表中，星号 (*) 表示核编号（值为 0 和 1）。

示例：Port_NAME_*

- Port_NAME_0: 表示第一个核
- Port_NAME_1: 表示第二个核（将核数指定为 2 时显示）

公共时钟/复位信号

表 284：公共时钟/复位信号

名称	大小	I/O	描述
sys_reset	1	输入	核的异步复位 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
dclk	1	输入	稳定/自由运行的输入时钟，连接到 GT 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_refclk_p	1	输入	差分输入时钟，连接到 GT。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
gt_refclk_n	1	输入	差分输入时钟，连接到 GT。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
qpll0_clk_in_*	2/4	输入	QPLL0 时钟输入。（QPPL 是四通道锁相环） 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。 端口宽度：50G 单核为 2 位，40G 单核和 50G 双核为 4 位。
qpll0_refclk_in_*	2/4	输入	QPLL0 参考时钟输入。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。 端口宽度：50G 单核为 2 位，40G 单核和 50G 双核为 4 位。
qpll1_clk_in_*	2/4	输入	QPLL1 时钟输入。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。 端口宽度：50G 单核为 2 位，40G 单核和 50G 双核则为 4 位。
qpll1_refclk_in_*	2/4	输入	QPLL1 参考时钟输入。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。端口宽度：50G 单核为 2 位，40G 单核和 50G 双核则为 4 位。
gtwiz_reset_qpll0_lock_in_*	1	输入	QPLL0 锁定复位输入，连接到 GT。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gtwiz_reset_qpll0_reset_out_*	1	输出	QPLL0 锁定复位输出，来自 GT。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
tx_clk_out_*	1	输出	来自 GT 的 TX 用户时钟输出。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
rx_serdes_clk_*	1	输入	RX SerDes 时钟输入，连接到核 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。GT 类型并非 GTM。
rxreclkout_*	1	输出	来自 GT 的 RX 恢复时钟输出。

表 284：公共时钟/复位信号 (续)

名称	大小	I/O	描述
tx_core_clk_*	1	输入	来自 GT 封装器的 TX 核时钟输入。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
rx_core_clk_*	1	输入	RX 核时钟输入，连接到核。
tx_mii_clk_*	1	输出	来自 GT 的 TX 用户时钟输出。 在“GT Selection and Configuration”选项卡下所选核类型为“Ethernet MAC+PCS/PMA”且选中“Include GT subcore in core”选项时，此端口可用。
rx_clk_out_*	1	输出	来自 GT 的 RX 用户时钟输出。
tx_reset_*	1	输入	TX 复位输入，连接到核。
user_tx_reset_*	1	输出	对应用户逻辑的 TX 复位输出。当时钟变为不稳定时，user_tx_reset 将转至高电平以复位用户逻辑。当 GT resetdone 转至低电平时，此信号将断言有效。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
gt_reset_tx_done_out_*	1	输出	来自 GT 的 TX 复位已完成信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
rx_reset_*	1	输入	RX 复位输入，连接到核。
user_rx_reset_*	1	输出	对应用户逻辑的 RX 复位输出。当时钟变为不稳定时，user_rx_reset 将转至高电平以复位用户逻辑。当 GT resetdone 转至低电平时，此信号将断言有效。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
rsfec_txoutclk_out_*	1	输入	作为软核 KP4 RS-FEC 块的稳定 TX 输入时钟。 注释： 在“Shared Logic”选项卡下启用“Soft KP4 RS-FEC”选项并选中“Include Shared Logic in Example Design”选项时，此端口可供 GTM 器件使用。您必须向 KP4 RS-FEC 块提供此时钟信号（频率为 332.03125 MHz）。
rsfec_rxoutclk_out_*	1	输入	作为软核 KP4 RS-FEC 块的稳定 RX 输入时钟。 注释： 在“Shared Logic”选项卡下启用“Soft KP4 RS-FEC”选项并选中“Include Shared Logic in Example Design”选项时，此端口可供 GTM 器件使用。您必须向 KP4 RS-FEC 块提供此时钟信号（频率为 332.03125 MHz）。
rsfec_tx_locked_*	1	输入	作为软核 KP4 RS-FEC 块的稳定 TX resetdone。 注释： 在“Shared Logic”选项卡下启用“Soft KP4 RS-FEC”选项并选中“Include Shared Logic in Example Design”选项时，此端口可供 GTM 器件使用。您必须向 KP4 RS-FEC 块提供此信号。
rsfec_rx_locked_*	1	输入	作为软核 KP4 RS-FEC 块的稳定 RX Resetdone。 注释： 在“Shared Logic”选项卡下启用“Soft KP4 RS-FEC”选项并选中“Include Shared Logic in Example Design”选项时，此端口可供 GTM 器件使用。您必须向 KP4 RS-FEC 块提供此信号。

表 284：公共时钟/复位信号 (续)

名称	大小	I/O	描述
gt_reset_rx_done_out_*	1	输出	来自 GT 的 RX 复位已完成信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in core”时，此端口可用。
rx_serdes_reset_*	1	输入	RX SerDes 复位信号。 在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项时，此端口可用。GT 类型并非 GTM。
ctl_gt_reset_all_*	1	输出	来自 AXI4-Lite 寄存器映射的 gt_reset_all 信号。 在“Configuration”选项卡下选中“Include AXI4-Lite”并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gtwiz_reset_all_in_*	1	输入	来自用户的 gt_reset_all 信号。 在“Configuration”选项卡下选中“Control and Statistics Interface”时，此端口可用。
ctl_gt_tx_reset_*	1	输出	来自 AXI4-Lite 寄存器映射的 gt_tx_reset 信号。 在“Configuration”选项卡下选中“Include AXI4-Lite”并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gtwiz_tx_datapath_reset_in_*	1	输入	来自用户的 gt_tx_reset 信号。 在“Configuration”选项卡下选中“Control and Statistics Interface”时，此端口可用。
ctl_gt_rx_reset_*	1	输出	来自 AXI4-Lite 寄存器映射的 gt_rx_reset 信号。 在“Configuration”选项卡下选中“Include AXI4-Lite”并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gtwiz_rx_datapath_reset_in_*	1	输入	来自用户的 gt_rx_reset 信号。 在“Configuration”选项卡下选中“Control and Statistics Interface”时，此端口可用。
gt_reset_all_in_*	1	输入	来自共享逻辑封装器的 reset_wrapper 的 gt_reset_all 信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gt_tx_reset_in_*	1	输入	来自共享逻辑封装文件的 reset_wrapper 的 gt_tx_reset_in 信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gt_rx_reset_in_*	1	输入	来自共享逻辑封装器的 reset_wrapper 的 gt_rx_reset_in 信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项并在“Shared Logic”选项卡下选中“Include Shared Logic in example design”时，此端口可用。
gt_refclk_out	1	输出	gt_refclk 与 gt_ref_clk 相同，用来驱动用户互连结构逻辑。
gtpowergood_out_*	2/4	输出	请参阅《UltraScale 架构 GTH 收发器用户指南》(UG576) 或《UltraScale 架构 GTY 收发器用户指南》(UG578) 以获取端口描述。
TXOUTCLKSEL_IN_*	6/12	输入	此端口用于选择 gtwizard TX 输出时钟的时钟源。此端口根据预置以 6'b101101/12'b101101101101101 驱动。
RXOUTCLKSEL_IN_*	6/12	输入	此端口用于选择 gtwizard RX 输出时钟的时钟源。此端口根据预置以 6'b101101/12'b101101101101101 驱动。
gtwiz_reset_all_in*	1	输入	来自用户的 gt_reset_all 信号。 注释： 仅适用于 Versal 器件。在“Configuration”选项卡下选中“Control and Statistics interface”时，此端口可用。

表 284：公共时钟/复位信号 (续)

名称	大小	I/O	描述
gtwiz_rx_datapath_reset_in_*	1	输入	来自用户的 gt_rx_reset 信号。 注释： 仅适用于 Versal 器件。在“Configuration”选项卡下选中“Control and Statistics interface”时，此端口可用。
gtm_txusrclk_out2_*	1	输出	表示核中包含 GT 时，来自该核的 TX 时钟输出。 注释： 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项、在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项，且所选“GT Type”为 GTM 时，此端口可用。
gtm_rxusrclk_out2_*	1	输出	表示核中包含 GT 时，来自该核的 RX 时钟输出。 注释： 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项、在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项，且所选“GT Type”为 GTM 时，此端口可用。
gtm_out_txprgdivresetdone_*	1	输出	表示核中包含 GT 时，来自该核的 TX 复位输出。 注释： 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项、在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项，且所选“GT Type”为 GTM 时，此端口可用。
gtm_out_rxprgdivresetdone_*	1	输出	表示核中包含 GT 时，来自该核的 RX 复位输出。 注释： 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项、在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项，且所选“GT Type”为 GTM 时，此端口可用。
txoutclk_out_*	1	输入	TX 时钟输入，连接到核。 注释： 在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项时，此端口可供 GTM 器件使用（非切换配置）。您必须向该核提供此信号（针对 40G 提供的频率为 312.5 MHz，针对 50G 提供的频率为 390.625 MHz）。
rxoutclk_out_*	1	输入	RX 时钟输入，连接到核。 注释： 在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项时，此端口可供 GTM 器件使用（非切换配置）。您必须向该核提供此频率（针对 40G 提供的频率为 312.5 MHz，针对 50G 提供的频率为 390.625 MHz）。
tx_locked_*	1	输入	TX resetdone 输入，连接到核。 注释： 在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项时，此端口可供 GTM 器件使用。您必须按照 txoutclk_out_* 时钟向核提供此输入。
rx_locked_*	1	输入	RX resetdone 输入，连接到核。 注释： 在“Shared Logic”选项卡下选中“Include Shared Logic in example design”选项时，此端口可供 GTM 器件使用。您必须按照 rxoutclk_out_* 时钟向核提供此输入。

表 284：公共时钟/复位信号 (续)

名称	大小	I/O	描述
txoutclk_out_50	1	输入	TX 时钟输入，连接到核。 注释： 此端口可供 Versal GTM 器件使用（适用于切换配置）。
txoutclk_out_40	1	输入	TX 时钟输入，连接到核。 注释： 此端口可供 Versal GTM 器件使用（适用于切换配置）。
rxoutclk_out_50	1	输入	注释： 此端口可供 Versal GTM 器件使用（适用于切换配置）。
rxoutclk_out_40	1	输入	RX 时钟输入，连接到核。 注释： 此端口可供 Versal GTM 器件使用（适用于切换配置）。

适用于 UltraScale/UltraScale+ 器件的常用收发器端口

表 285：适用于 UltraScale/UltraScale+ 器件的常用收发器端口

名称	大小	I/O	描述
gt_loopback_in	6/12	输入	GT 环回输入信号。请参阅 GT 用户指南。 针对 50G 单核，位宽为 6 位，针对 40G 单核或 50G 双核，并在“GT Selection and Configuration”（GT 选择和配置）选项卡下选中“Include GT subcore in core”（在核中包含 GT 子核）选项时，位宽为 12 位。
gt_rxp_in_0	1	输入	差分串行 GT RX 输入（针对通道 0）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_rxn_in_0	1	输入	差分串行 GT RX 输入（针对通道 0）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_rxp_in_1	1	输入	差分串行 GT RX 输入（针对通道 1）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_rxn_in_1	1	输入	差分串行 GT RX 输入（针对通道 1）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_rxp_in_2	1	输入	差分串行 GT RX 输入（针对通道 2）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_rxn_in_2	1	输入	差分串行 GT RX 输入（针对通道 2）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_rxp_in_3	1	输入	差分串行 GT RX 输入（针对通道 3）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。

表 285：适用于 UltraScale/UltraScale+ 器件的常用收发器端口 (续)

名称	大小	I/O	描述
gt_rxn_in_3	1	输入	差分串行 GT RX 输入（针对通道 3）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_txp_out_0	1	输出	差分串行 GT TX 输出（针对通道 0）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_txn_out_0	1	输出	差分串行 GT TX 输出（针对通道 0）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_txp_out_1	1	输出	差分串行 GT TX 输出（针对通道 1）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_txn_out_1	1	输出	差分串行 GT TX 输出（针对通道 1）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用。
gt_txp_out_2	1	输出	差分串行 GT TX 输出（针对通道 2）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_txn_out_2	1	输出	差分串行 GT TX 输出（针对通道 2）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_txp_out_3	1	输出	差分串行 GT TX 输出（针对通道 3）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_txn_out_3	1	输出	差分串行 GT TX 输出（针对通道 3）。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in core”选项时，此端口可用于 40G 和 50G 双核。
gt_txp_out	1/2	输出	差分串行 GT TX 输出 此端口可用于获取开发板支持。
gt_txn_out	1/2	输出	差分串行 GT TX 输出 此端口可用于获取开发板支持。
gt_txp_in	1/2	输出	差分串行 GT TX 输入 此端口可用于获取开发板支持。
gt_txn_in	1/2	输入	差分串行 GT TX 输入 此端口可用于获取开发板支持。
gt_loopback_out_*	1	输出	GT 环回输出信号（来自 AXI4-Lite 寄存器映射）。请参阅相应的 GT 用户指南。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并在“Configuration”选项卡下选中 AXI4-Lite 接口时，此端口可用。
rxgearboxslip_out_*	1	输出	从核到 GT 的 Rxgearboxslip 信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。

表 285：适用于 UltraScale/UltraScale+ 器件的常用收发器端口 (续)

名称	大小	I/O	描述
rxdatavalid_in_*	2/4	输入	从 GT 到核的 RX 数据有效信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
rxheader_in_*	6/12	输入	从 GT 到核的 RX 报头信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
rxheadervvalid_in_*	2/4	输入	从 GT 到核的 RX 报头有效信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
rx_serdes_data_in_*	255	输入	从核到 GT 的 TX 数据信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
txheader_out_*	6/12	输出	从核到 GT 的 TX 报头信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
tx_serdes_data_out_*	255	输出	从核到 GT 的 TX 数据信号。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
txsequence_out_*	7	输出	从核到 GT 的 TX 顺序信号。

适用于 Versal ACAP 的常用收发器端口

表 286：适用于 Versal ACAP 的常用收发器端口

名称	大小	I/O	描述
gtwiz_loopback_*	3	输出	GT 环回输出信号（来自 AXI4-Lite 寄存器映射）。请参阅相应的 GT 用户指南。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并在“Configuration”选项卡下选中 AXI4-Lite 接口时，此端口可用。 注释： 在 bd 设计中，您需要手动连接此信号。
gtwiz_tx_rate_*	8	输出	可从 AXI4-Lite 寄存器映射中选择 GT TX 线速率。请参阅相应的 GT 用户指南。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并在“Configuration”选项卡下选中 AXI4-Lite 接口时，此端口可用。
gtwiz_rx_rate_*	8	输出	从 AXI4-Lite 寄存器映射选中的 GT TX 线速率。请参阅相应的 GT 用户指南。 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并在“Configuration”选项卡下选中 AXI4-Lite 接口时，此端口可用。

表 286：适用于 Versal ACAP 的常用收发器端口 (续)

名称	大小	I/O	描述
tx_serdes_data_core0_lane0_out	128	输出	如果 GT 类型并非 GTM，那么这表示 Lane0 的 TX 数据输出信号（从核到 GT）；对于 50G，有效位是下 64 位，对于 40G 速度，有效位是下 32 位。
tx_serdes_data_core0_lane1_out	128	输出	如果 GT 类型并非 GTM，那么这表示 Lane1 的 TX 数据输出信号（从核到 GT）；对于 50G，有效位是下 64 位，对于 40G 速度，有效位是 32 位。
tx_serdes_data_core0_lane2_out	128	输出	如果 GT 类型并非 GTM，那么这表示 Lane2 的 TX 数据输出信号（从核到 GT）；对于 40G 速度，有效位是下 32 位。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_serdes_data_core0_lane3_out	128	输出	如果 GT 类型并非 GTM，那么这表示 Lane3 的 TX 数据输出信号（从核到 GT）；对于 40G 速度，有效位是下 32 位。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_serdes_header_core0_lane0_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane0 的 TX 报头输出信号（从核到 GT）
tx_serdes_header_core0_lane1_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane1 的 TX 报头输出信号（从核到 GT）
tx_serdes_header_core0_lane2_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane2 的 TX 报头输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_serdes_header_core0_lane3_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane3 的 TX 报头输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_data_core0_lane0_in	128	输入	如果 GT 类型并非 GTM，那么这表示 Lane0 的 RX 数据输入信号（从 GT 到核）；对于 50G，有效位是下 64 位，对于 40G 速度，有效位是下 32 位。
rx_serdes_data_core0_lane1_in	128	输入	如果 GT 类型并非 GTM，那么这表示 Lane1 的 RX 数据输入信号（从 GT 到核）；对于 50G，有效位是下 64 位，对于 40G 速度，有效位是下 32 位。
rx_serdes_data_core0_lane2_in	128	输入	如果 GT 类型并非 GTM，那么这表示 Lane2 的 RX 数据输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_data_core0_lane3_in	128	输入	如果 GT 类型并非 GTM，那么这表示 Lane3 的 RX 数据输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_header_core0_lane0_in	6	输入	如果 GT 类型并非 GTM，那么这表示 Lane0 的 RX 报头输入信号（从 GT 到核）。
rx_serdes_header_core0_lane1_in	6	输入	如果 GT 类型并非 GTM，那么这表示 Lane1 的 RX 报头输入信号（从 GT 到核）。
rx_serdes_header_core0_lane2_in	6	输入	如果 GT 类型并非 GTM，那么这表示 Lane2 的 RX 报头输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。

表 286：适用于 Versal ACAP 的常用收发器端口 (续)

名称	大小	I/O	描述
rx_serdes_header_core0_lane3_in	6	输入	如果 GT 类型并非 GTM，那么这表示 Lane3 的 RX 报头输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_serdes_seq_core0_lane0_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane0 的 TX 序列输出信号（从核到 GT）。
tx_serdes_seq_core0_lane1_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane1 的 TX 序列输出信号（从核到 GT）。
tx_serdes_seq_core0_lane2_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane2 的 TX 序列输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_serdes_seq_core0_lane3_out	6	输出	如果 GT 类型并非 GTM，那么这表示 Lane3 的 TX 序列输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_headervalid_core0_lane0	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane0 的 RX 报头有效输入信号（从 GT 到核）。
rx_serdes_headervalid_core0_lane1	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane1 的 RX 报头有效输入信号（从 GT 到核）。
rx_serdes_headervalid_core0_lane2	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane2 的 RX 报头有效输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_headervalid_core0_lane3	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane3 的 RX 报头有效输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_datavalid_core0_lane0	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane0 的 RX 数据有效输入信号（从 GT 到核）。
rx_serdes_datavalid_core0_lane1	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane1 的 RX 数据有效输入信号（从 GT 到核）。
rx_serdes_datavalid_core0_lane2	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane2 的 RX 数据有效输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_datavalid_core0_lane3	2	输入	如果 GT 类型并非 GTM，那么这表示 Lane3 的 RX 数据有效输入信号（从 GT 到核）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_bitslip_core0_lane0	1	输出	如果 GT 类型并非 GTM，那么这表示 Lane0 的 RX 变速箱输出信号（从核到 GT）。
rx_serdes_bitslip_core0_lane1	1	输出	如果 GT 类型并非 GTM，那么这表示 Lane1 的 RX 变速箱输出信号（从核到 GT）。
rx_serdes_bitslip_core0_lane2	1	输出	如果 GT 类型并非 GTM，那么这表示 Lane2 的 RX 变速箱输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_serdes_bitslip_core0_lane3	1	输出	如果 GT 类型并非 GTM，那么这表示 Lane3 的 RX 变速箱输出信号（从核到 GT）。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。

表 286：适用于 Versal ACAP 的常用收发器端口 (续)

名称	大小	I/O	描述
mst_tx_resetdone_0	1	输入	从 GT 到核的 TX 主 resetdone 信号，用于指示 lane0 状态。
mst_rx_resetdone_0	1	输入	从 GT 到核的 RX 主 resetdone 信号，用于指示 lane0 状态。
mst_tx_resetdone_1	1	输入	从 GT 到核的 TX 主 resetdone 信号，用于指示 lane1 状态。
mst_rx_resetdone_1	1	输入	从 GT 到核的 RX 主 resetdone 信号，用于指示 lane1 状态。
mst_tx_resetdone_2	1	输入	从 GT 到核的 TX 主 resetdone 信号，用于指示 lane2 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
mst_rx_resetdone_2	1	输入	从 GT 到核的 RX 主 resetdone 信号，用于指示 lane2 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
mst_tx_resetdone_3	1	输入	从 GT 到核的 TX 主 resetdone 信号，用于指示 lane3 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
mst_rx_resetdone_3	1	输入	从 GT 到核的 RX 主 resetdone 信号，用于指示 lane3 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_pma_resetdone_0	1	输入	从 GT 到核的 TX PMA resetdone 信号，用于指示 lane0 状态。
rx_pma_resetdone_0	1	输入	从 GT 到核的 RX PMA resetdone 信号，用于指示 lane0 状态。
tx_pma_resetdone_1	1	输入	从 GT 到核的 TX PMA resetdone 信号，用于指示 lane1 状态。
rx_pma_resetdone_1	1	输入	从 GT 到核的 RX PMA resetdone 信号，用于指示 lane1 状态。
tx_pma_resetdone_2	1	输入	从 GT 到核的 TX PMA resetdone 信号，用于指示 lane2 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_pma_resetdone_2	1	输入	从 GT 到核的 RX PMA resetdone 信号，用于指示 lane2 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
tx_pma_resetdone_3	1	输入	从 GT 到核的 TX PMA resetdone 信号，用于指示 lane3 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
rx_pma_resetdone_3	1	输入	从 GT 到核的 RX PMA resetdone 信号，用于指示 lane3 状态。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
mst_tx_reset_0	1	输出	Lane0 的 TX 主控制器复位输出信号（从核到 GT）。
mst_rx_reset_0	1	输出	Lane0 的 RX 主控制器复位输出信号（从核到 GT）。
mst_tx_reset_1	1	输出	Lane1 的 TX 主控制器复位输出信号（从核到 GT）。

表 286：适用于 Versal ACAP 的常用收发器端口 (续)

名称	大小	I/O	描述
mst_rx_reset_1	1	输出	Lane1 的 RX 主控制器复位输出信号 (从核到 GT)。
mst_tx_reset_2	1	输出	Lane2 的 TX 主控制器复位输出信号 (从核到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
mst_rx_reset_2	1	输出	Lane2 的 RX 主控制器复位输出信号 (从核到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
mst_tx_reset_3	1	输出	Lane3 的 TX 主控制器复位输出信号 (从核到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
mst_rx_reset_3	1	输出	Lane3 的 RX 主控制器复位输出信号 (从核到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
txuserddy_out_0	1	输出	Lane0 的 TX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。
rxuserddy_out_0	1	输出	Lane0 的 RX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。
txuserddy_out_1	1	输出	Lane1 的 TX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。
rxuserddy_out_1	1	输出	Lane1 的 RX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。
txuserddy_out_2	1	输出	Lane2 的 TX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
rxuserddy_out_2	1	输出	Lane2 的 RX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
txuserddy_out_3	1	输出	Lane3 的 TX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
rxuserddy_out_3	1	输出	Lane3 的 RX 用户就绪输出信号 (从核 (复位接口 IP) 到 GT)。 仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时, 此端口才可用。
rx_resetdone_out_0	1	输出	从核到设计示例的 RX 用户就绪输出信号。
tx_resetdone_out_0	1	输出	从核到设计示例的 TX 用户就绪输出信号。
mst_rx_dp_reset_*	1	输出	从 GT 复位 IP 到核的 RX 复位输出信号 (仅限 Versal 器件)。
mst_tx_dp_reset_*	1	输出	从 GT 复位 IP 到核的 TX 复位输出信号 (仅限 Versal 器件)。
gtm_txdata_in_0	255	输出	如果 GT 类型为 GTM, 那么这表示 Lane0 的 TX 数据输出信号 (从核到 GT)。
gtm_txdata_in_1	255	输出	如果 GT 类型为 GTM, 那么这表示 Lane1 的 TX 数据输出信号 (从核到 GT)。

表 286：适用于 Versal ACAP 的常用收发器端口 (续)

名称	大小	I/O	描述
gtm_txdata_in_2	255	输出	如果 GT 类型为 GTM，那么这表示 Lane2 的 TX 数据输出信号（从核到 GT）。仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
gtm_txdata_in_3	255	输出	如果 GT 类型为 GTM，那么这表示 Lane3 的 TX 数据输出信号（从核到 GT）。仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
gtm_rxdata_out_0	255	输入	如果 GT 类型为 GTM，那么这表示 Lane0 的 RX 数据输入信号（从 GT 到核）
gtm_rxdata_out_1	255	输入	如果 GT 类型为 GTM，那么这表示 Lane1 的 RX 数据输入信号（从 GT 到核）
gtm_rxdata_out_2	255	输入	如果 GT 类型为 GTM，那么这表示 Lane2 的 RX 数据输入信号（从 GT 到核）。仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。
gtm_rxdata_out_3	255	输入	如果 GT 类型为 GTM，那么这表示 Lane3 的 RX 数据输入信号（从 GT 到核）。仅当核速度为 40G 或者在“Configuration”选项卡下选中“Runtime Switchable”选项时，此端口才可用。

收发器控制调试端口和状态调试端口

选中“Enable Additional GT Control/Status and DRP Ports”（启用其它 GT 控制/状态和 DRP 端口）并在“GT Selection and Configuration”（GT 选择和配置）选项卡下选中“GT in Core”（核内 GT）选项时，本节下所述端口可用。请参阅 GT 用户指南以获取端口描述。

表 287：收发器控制调试端口和状态调试端口

名称	大小	I/O	描述
gt_dmonitorout_*	34/68	输出	端口宽度：针对 50G 单核为 34 位，针对 40G 则为 68 位。
gt_eyes candataerror_*	2/4	输出	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_eyes canreset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_eyes cantrigger_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_pcsrsvdin_*	32/64	输入	端口宽度：针对 50G 单核为 32 位，针对 40G 则为 64 位。
gt_rxbu freset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxbu fstatus_*	6/12	输出	端口宽度：针对 50G 单核为 6 位，针对 40G 则为 12 位。
gt_rxcd rhold_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxcomm adeten_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxdfeagchold_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxdfel p mreset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxl atclk_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxl pmen_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxp csreset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxp mareset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxpolarity_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。

表 287：收发器控制调试端口和状态调试端口 (续)

名称	大小	I/O	描述
gt_rxprbscntreset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxprbserr_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxprbsssel_*	8/16	输入	端口宽度：针对 50G 单核为 8 位，针对 40G 则为 16 位。
gt_rxrate_*	6/12	输入	端口宽度：针对 50G 单核为 6 位，针对 40G 则为 12 位。
gt_rxslide_in_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_rxstartofseq_*	4/8	输出	端口宽度：针对 50G 单核为 4 位，针对 40G 则为 8 位。
gt_txbufstatus_*	4/8	输出	端口宽度：针对 50G 单核为 4 位，针对 40G 则为 8 位。
gt_txdiffctrl_*	10/20	输入	端口宽度：针对 50G 单核为 10 位，针对 40G 则为 20 位。
gt_txinhibit_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txlatclk_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txmaincursor_*	14/28	输入	端口宽度：针对 50G 单核为 14 位，针对 40G 则为 28 位。
gt_txpcreset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txpmareset_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txpolarity_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txpostcursor_*	10/20	输入	端口宽度：针对 50G 单核为 10 位，针对 40G 则为 20 位。
gt_txprbsforceerr_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_txprbsssel_*	8/16	输入	端口宽度：针对 50G 单核为 8 位，针对 40G 则为 16 位。
gt_txprecursor_*	10/20	输入	端口宽度：针对 50G 单核为 10 位，针对 40G 则为 20 位。
gtwiz_reset_tx_datapath_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gtwiz_reset_rx_datapath_*	2/4	输入	端口宽度：针对 50G 单核为 2 位，针对 40G 则为 4 位。
gt_common_drpcclk	1	输入	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drpdo	16	输出	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drprdy	1	输出	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drpen	1	输入	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drpwe	1	输入	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drpaddr	10	输入	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_common_drpdi	16	输入	在“Shared Logic”选项卡下选中“Include Shared Logic in core”选项时，此端口可用。
gt_ch_drpcclk_0	1	输入	
gt_ch_drpdo_0	16	输出	
gt_ch_drprdy_0	1	输出	
gt_ch_drpen_0	1	输入	
gt_ch_drpwe_0	1	输入	
gt_ch_drpaddr_0	10	输入	
gt_ch_drpdi_0	16	输入	
gt_ch_drpcclk_1	1	输入	

表 287：收发器控制调试端口和状态调试端口 (续)

名称	大小	I/O	描述
gt_ch_drpdo_1	16	输出	
gt_ch_drprdy_1	1	输出	
gt_ch_drpen_1	1	输入	
gt_ch_drpwe_1	1	输入	
gt_ch_drpaddr_1	10	输入	
gt_ch_drpdi_1	16	输入	
gt_ch_drpclk_2	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpdo_2	16	输出	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drprdy_2	1	输出	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpen_2	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpwe_2	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpaddr_2	10	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpdi_2	16	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpclk_3	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpdo_3	16	输出	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drprdy_3	1	输出	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpen_3	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpwe_3	1	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpaddr_3	10	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。
gt_ch_drpdi_3	16	输入	当核速度为 40G 或者 50G (双核) 时, 此端口可用。

AXI4-Lite 接口端口

在“Configuration”（配置）选项卡下选中“Include AXI4-Lite”时，本节下所述端口可用。

表 288：AXI4-Lite 接口端口

名称	大小	I/O	描述
s_axi_aclk_*	1	输入	AXI 时钟信号
s_axi_aresetn_*	1	输入	AXI 复位信号
pm_tick_*	1	输入	PM 时钟节拍用户输入
s_axi_awaddr_*	32	输入	AXI 写入地址
s_axi_awvalid_*	1	输入	AXI 写入地址有效
s_axi_awready_*	1	输出	AXI 写入地址就绪
s_axi_wdata_*	32	输入	AXI 写入数据
s_axi_wstrb_*	4	输入	AXI 写入选通。此信号用于指示保存有效数据的字节通道。
s_axi_wvalid_*	1	输入	AXI 写入数据有效。此信号用于指示有效的写入数据和写入选通均可供使用。
s_axi_wready_*	1	输出	AXI 写入数据就绪
s_axi_bresp_*	2	输出	AXI 写入响应。此信号用于指示写入传输事务的状态。 'b00 = OKAY 'b01 = EXOKAY 'b10 = SLVERR 'b11 = DECERR

表 288: AXI4-Lite 接口端口 (续)

名称	大小	I/O	描述
s_axi_bvalid_*	1	输出	AXI 写入响应有效。此信号用于指示该通道正在发出有效的写入响应信号。
s_axi_bready_*	1	输入	AXI 写入响应就绪。
s_axi_araddr_*	32	输入	AXI 读取地址
s_axi_arvalid_*	1	输入	AXI 读取地址有效
s_axi_arready_*	1	输出	AXI 读取地址就绪
s_axi_rdata_*	32	输出	AXI 读取数据 (由从接口发出)
s_axi_rresp_*	2	输出	AXI 读取响应。此信号用于指示读取传输的状态。 'b00 = OKAY 'b01 = EXOKAY 'b10 = SLVERR 'b11 = DECERR
s_axi_rvalid_*	1	输出	AXI 读取数据有效
s_axi_rready_*	1	输入	AXI 读取就绪。此信号用于指示用户/主控制器可接受读取数据和响应信息。

128 位跨接式 AXI4-Stream 用户接口信号

在“Configuration”选项卡下选中“Ethernet MAC+PCS/PMA with the 128-bit Straddle Packet AXI4-Stream”选项时，本节下所述端口可用。

表 289: 128 位跨接式 AXI4-Stream 用户接口信号

名称	大小	I/O	描述
tx_unfout_*	1	输出	来自核的 TX 路径下溢信号。如果 tx_unfout_* 采样为 1，则表示已发生违例，即当前包已受损。只要下溢状况仍然存在，就会发射错误控制块信号。 用户逻辑负责将完整的包输入核，并避免 TX 数据路径接口欠载。 注释： 当此信号采样为 1 时，您必须应用 tx_reset/sys_reset 才能将核从下溢问题恢复。tx_reset 仅复位 TX 路径，sys_reset 则会恢复整个系统。
tx_axis_tready_*	1	输出	来自核的 TX 路径就绪信号。
tx_axis_tvalid_*	1	输入	发射 AXI4-Stream 数据有效。
tx_axis_tdata_*	128	输入	发射 AXI4-Stream 数据总线。

表 289: 128 位跨接式 AXI4-Stream 用户接口信号 (续)

名称	大小	I/O	描述
tx_axis_tuser_*	70	输入	TX 段和包信息信号。tx_axis_tuser_0[69:0]
			69 - tx_axis_tuser_err1
			68:66 - tx_axis_tuser_mty1[2:0]
			65 - tx_axis_tuser_eop1
			64 - tx_axis_tuser_sop1
			63 - tx_axis_tuser_ena1
			62 - tx_axis_tuser_err0
			61:59 - tx_axis_tuser_mty0[2:0]
			58 - tx_axis_tuser_eop0
			57 - tx_axis_tuser_sop0
			56 - tx_axis_tuser_ena0
			55:0 - tx_preamblein
rx_axis_tvalid_*	1	输出	接收 AXI4-Stream 数据有效。
rx_axis_tdata_*	128	输出	接收 AXI4-Stream 数据总线。
rx_axis_tuser	70	输出	RX 段和包信息信号。rx_axis_tuser_0[69:0]
			69 - rx_axis_tuser_err1
			68:66 - rx_axis_tuser_mty1[2:0]
			65 - rx_axis_tuser_eop1
			64 - rx_axis_tuser_sop1
			63 - rx_axis_tuser_ena1
			62 - rx_axis_tuser_err0
			61:59 - rx_axis_tuser_mty0[2:0]
			58 - rx_axis_tuser_eop0
			57 - rx_axis_tuser_sop0
			56 - rx_axis_tuser_ena0
			55:0 - rx_preamblein

256 位 AXI4-Stream 端口

在“Configuration”选项卡下选中“Ethernet MAC + PCS/PMA with the 256-bit AXI4-Stream”选项时，本节下所述端口可用。

表 290: 256 位 AXI4-Stream 端口

名称	大小	I/O	描述
tx_axis_tready_*	1	输出	AXI4-Stream 确认信号，用于指示开始数据传输。
tx_axis_tvalid_*	1	输入	AXI4-Stream 数据有效输入
tx_axis_tdata_*	256	输入	AXI4-Stream 数据
tx_axis_tuser_*	1	输入	AXI4-Stream 用户边带接口。 1 指示接收到的包无效。 0 指示接收到的包有效。

表 290: 256 位 AXI4-Stream 端口 (续)

名称	大小	I/O	描述
tx_axis_tlast_*	1	输入	AXI4-Stream 信号, 指示以太网包结束
tx_axis_tkeep_*	32	输入	AXI4-Stream 数据控制
rx_axis_tdata_*	256	输出	AXI4-Stream 数据, 可连接到用户逻辑。
rx_axis_tvalid_*	1	输出	AXI4-Stream 数据有效。当此信号设为 1 时, RX AXI 总线上存在有效数据。
rx_axis_tuser_*	1	输出	AXI4-Stream 用户边带接口。 1 指示接收到的包无效。 0 指示接收到的包有效。
rx_axis_tlast_*	1	输出	AXI4-Stream 信号, 指示包结束。
rx_axis_tkeep_*	32	输出	AXI4-Stream 数据控制, 连接到上层。

MII 用户接口信号

在“Configuration”（配置）选项卡下选中“Ethernet PCS/PMA”（以太网 PCS/PMA）时, 本节下所述端口可用。

表 291: MII 用户接口信号

名称	大小	I/O	描述
tx_mii_d_*	128	输入	发射 XLGMII/50GMII 数据总线。
tx_mii_c_*	16	输入	XLGMII/50GMII 控制总线。
rx_mii_d_*	128	输出	接收 XLGMII/50GMII 数据总线。
rx_mii_c_*	16	输出	XLGMII/50GMII 接收控制总线。

TX 路径控制信号、状态信号和统计数据信号

表 292: TX 路径控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_tx_test_pattern_*	1	输入	针对 TX 核启用测试模式生成。值为 1 即表示根据第 82.2.10 条中的定义启用测试模式。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.42.7。生成扰码空闲模式。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_tx_enable_*	1	输入	TX 使能 (启用)。此信号采样为 1 时用于启用数据发射。当此信号采样为 0 时, 核仅发射空闲信号。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
ctl_tx_fcs_ins_enable_*	1	输入	启用由 TX 核执行 FCS 插入。如果该位设为 0, 40G/50G High Speed Ethernet 子系统不会向数据包添加 FCS。如果该位设为 1, 那么 40G/50G High Speed Ethernet Subsystem 会计算 FCS 并将其添加到数据包中。在不同数据包之间不得动态更改此输入。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。

表 292：TX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_tx_ipg_value_*	4	输入	此信号为可选。ctl_tx_ipg_value 用于定义在 rx_serdes_clk 包之间插入的目标平均最小包间隔 (IPG, 以字节为单位)。有效值范围为 8 到 12。ctl_tx_ipg_value 还可编程为范围介于 0 到 7 之间的值, 但在此情况下, 它解释为表示“最小 IPG”, 因此仅插入终止 (Terminate) 代码字 IPG; 在此情况下从不添加空闲 (Idle), 因此发射随机大小的数据包时, 生成的 IPG 平均约为 4 字节。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 并禁用“Include FIFO Logic”时, 此端口可用。
ctl_tx_send_lfi_*	1	输入	发射本地故障指示 (LFI) 代码字。优先于 RFI。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
ctl_tx_send_rfi_*	1	输入	发射远端故障指示 (RFI) 代码字。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
ctl_tx_send_idle_*	1	输入	发射空闲代码字。如果此输入采样为 1, 那么 TX 路径仅发射空闲 (Idle) 代码字。当伙伴器件正在发送远端故障指示 (RFI) 代码字时, 此输入应设置为 1。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
ctl_tx_custom_preamble_enable_*	1	输入	当此信号断言有效时, 会启用使用 tx_preamblein 作为定制前导码, 而不是插入标准前导码。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 并禁用“Include FIFO Logic”时, 此端口可用。
ctl_tx_ignore_fcs_*	1	输入	启用由 TX 核在 AXI4-Stream 接口上执行 FCS 纠错。仅当 ctl_tx_fcs_ins_enable 为低电平时, 此输入才有效。如果此输入为低电平, 并且发射的数据包中含有无效 FCS, 则不会将其作为有效包丢弃。如果此输入为高电平, 则含无效 FCS 的包将被作为有效包丢弃。 在 stat_tx_bad_fcs 和 stomped_fcs 信号上会标记此错误, 并且此数据包将作为已接收的包进行发射。 报告的统计数据中显示不含 FCS 错误。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS 时, 此端口可用。
stat_tx_total_packets_*	1	输出	此信号为递增值, 对应已发射的数据包总数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_total_bytes_*	5	输出	此信号为递增值, 对应已发射的字节总数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_total_good_packets_*	1	输出	此信号为递增值, 对应已发射的有效数据包总数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_total_good_bytes_*	14	输出	此信号为递增值, 对应已发射的有效字节总数。仅当完全发射不含错误的数据包时, 该值才为非 0 值。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_packet_64_bytes_*	1	输出	此信号为递增值, 对应已发射的有效和无效数据包 (含 64 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_packet_65_127_bytes_*	1	输出	此信号为递增值, 对应已发射的有效和无效数据包 (含 65 到 127 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_packet_128_255_bytes_*	1	输出	此信号为递增值, 对应已发射的有效和无效数据包 (含 128 到 255 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_tx_packet_256_511_bytes_*	1	输出	此信号为递增值, 对应已发射的有效和无效数据包 (含 256 到 511 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。

表 292：TX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_tx_packet_512_1023_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 512 到 1,023 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_1024_1518_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 1,024 到 1,518 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_1519_1522_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 1,519 到 1,522 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_1523_1548_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 1,523 到 1,548 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_1549_2047_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 1,549 到 2,047 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_2048_4095_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 2,048 到 4,095 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_4096_8191_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 4,096 到 8,191 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_8192_9215_bytes_*	1	输出	此信号为递增值，对应已发射的有效和无效数据包（含 8,192 到 9,215 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_small_*	1	输出	此信号为递增值，对应所含数据小于 64 字节的所有包数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_packet_large_*	1	输出	此信号为递增值，对应所含数据超过 9,215 字节的所有包数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_bad_fcs_*	1	输出	此信号为递增值，对应具有 FCS 错误的数据包（大于 64 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_frame_error_*	1	输出	此信号为递增值，对应已设置 tx_errin（用于指示 EOP 中止）的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_tx_local_fault_*	1	输出	值为 1 表示接收解码器状态机当前处于 TX_INIT 状态。此输出对电平敏感。
stat_tx_fifo_error_*	1	输出	发射时钟补偿先入先出 (FIFO) 错误指示符。值为 1 表示时钟补偿 FIFO 发生下溢或上溢。仅当发射器时钟与局部参考时钟之间的 PPM 差超过 ± 200 ppm 时，才会发生此状况。 如果在任意时钟周期内此输出采样为 1，那么对应端口必须复位才能恢复正常运行。 当核类型为 Ethernet PCS/PMA 时，此端口可用。

RX 路径控制信号、状态信号和统计数据信号

表 293: RX 路径控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_rx_test_pattern_*	1	输入	针对 RX 核启用测试模式检查。值为 1 即表示根据第 82.2.17 条中的定义启用测试模式。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.42.2。检查扰码空闲模式。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_enable_*	1	输入	RX 使能（启用）。在正常操作期间，此输入必须设置为 1。当此输入设置为 0 时，RX 完成接收当前包（如果有）之后，就会停止接收包并阻止 PCS 继续对传入数据进行解码。在此模式下不报告任何统计数据，并且用户接口处于空闲状态。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_delete_fcs_*	1	输入	启用由 RX 核执行 FCS 移除。如果将此位设置为 0，则 40G/50G High Speed Ethernet 子系统不会移除传入包的 FCS。如果此位设置为 1，40G/50G High Speed Ethernet 子系统将会删除接收包的 FCS。对于长度小于等于 8 字节的包，不删除 FCS。仅当对应的复位输入断言有效时，才应更改此输入。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_ignore_fcs_*	1	输入	启用由 RX 核在用户接口上执行 FCS 纠错。如果此位设置为 0，对于接收到的含 FCS 错误的包，将随在上一次传输期间断言有效（rx_eopout 和 rx_enaout 均采样为 1）的 rx_errout 管脚一起发送。如果此位设置为 1，则 40G/50G High Speed Ethernet Subsystem 不会在用户接口处标记 FCS 错误。 报告的统计数据中显示此数据包有效。但 stat_rx_bad_fcs signa 信号会报告错误。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_max_packet_len_*	15	输入	长度超过该值的任何包都将被视为大小过大。如果包的大小大于该值，则该包将被截位至该值，并且 rx_errout 信号随 rx_eopout 信号一起断言有效。小于 16 字节的包将被丢弃。该总线允许的值范围为 64 到 16,383。 ctl_rx_max_packet_len[14] 保留并且必须设置为 0。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_min_packet_len_*	8	输入	长度小于该值的任何包都将被视为大小过小。如果包的大小小于该值，则在 rx_eopout 断言有效的周期内，rx_errout 信号将断言有效。小于 64 字节的包将被丢弃。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_custom_preamble_enable_*	1	输入	此信号断言有效时，会导致在 rx_preambleout 上呈现前导码。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 并禁用“Include FIFO Logic”时，此端口可用
ctl_rx_check_sfd_*	1	输入	此输入断言有效时，会使 Ethernet MAC 检查接收到的帧的帧起始定界符。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
ctl_rx_check_preamble_*	1	输入	此输入断言有效时，会使 Ethernet MAC 检查接收到的帧的前导码。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。

表 293: RX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_rx_process_lfi_*	1	输入	此输入设置为 1 时, RX 核会预测从 SerDes 传入的 LF 控制代码, 并对其进行处理。设置为 0 时, RX 核会忽略从 SerDes 传入的 LF 控制代码。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
ctl_rx_force_resync_*	1	输入	RX 强制再同步输入。此信号用于强制 RX 路径进行复位、再同步和重新对齐。值为 1 即强制执行复位操作。值为 0 允许正常操作。 注释: 此输入通常应为低电平 (Low) 并且仅限用于强制重新对齐时才应进行脉冲 (单周期最小脉冲)。 未选中 AXI4-Lite 接口且核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_block_lock_*	4	输出	每个 PCS 通道的块锁定状态。值为 1 表示对应通道已达到第 82 条中定义的块锁定状态。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.50.7:0 和 3.51.11:0。此输出对电平敏感。
stat_rx_framing_err_valid_0	1	输出	stat_rx_framing_err_0 的有效性指示符。当设为 1 时, 表示 stat_rx_framing_err_0 有效。
stat_rx_framing_err_0	3	输出	RX 同步报头位成帧错误。每个 PCS 通道都有 1 个 4 位总线, 用于指示针对该 PCS 通道接收到的同步报头错误的数量。仅当对应 stat_rx_framing_err_valid_0 设置为 1 时, 该总线的值才有效。这些总线上的值可随时更新, 并且这些值旨在作为递增值以供同步报头错误计数器使用。
stat_rx_framing_err_valid_1	1	输出	stat_rx_framing_err_1 的有效性指示符。当设为 1 时, 表示 stat_rx_framing_err_1 有效。
stat_rx_framing_err_1	3	输出	RX 同步报头位成帧错误。每个 PCS 通道都有 1 个 4 位总线, 用于指示针对该 PCS 通道接收到的同步报头错误的数量。仅当对应 stat_rx_framing_err_valid_1 设置为 1 时, 该总线的值才有效。这些总线上的值可随时更新, 并且这些值旨在作为递增值以供同步报头错误计数器使用。
stat_rx_framing_err_valid_2	1	输出	stat_rx_framing_err_2 的有效性指示符。当设为 1 时, 表示 stat_rx_framing_err_2 有效。
stat_rx_framing_err_2	3	输出	RX 同步报头位成帧错误。每个 PCS 通道都有 1 个 4 位总线, 用于指示针对该 PCS 通道接收到的同步报头错误的数量。仅当对应 stat_rx_framing_err_valid_2 设置为 1 时, 该总线的值才有效。这些总线上的值可随时更新, 并且这些值旨在作为递增值以供同步报头错误计数器使用。
stat_rx_framing_err_valid_3	1	输出	stat_rx_framing_err_3 的有效性指示符。当设为 1 时, 表示 stat_rx_framing_err_3 有效。
stat_rx_framing_err_3	3	输出	RX 同步报头位成帧错误。每个 PCS 通道都有 1 个 4 位总线, 用于指示针对该 PCS 通道接收到的同步报头错误的数量。仅当对应 stat_rx_framing_err_valid_3 设置为 1 时, 该总线的值才有效。这些总线上的值可随时更新, 并且这些值旨在作为递增值以供同步报头错误计数器使用。
stat_rx_vl_demuxed_*	4	输出	已找到 PCS 通道标记。如果该总线的信号采样为 1, 则表示接收器已对该 PCS 通道进行了正确的逆多路复用。 此输出对电平敏感。
stat_rx_vl_number_0	2	输出	该总线的值指示 PCS 通道 0 上出现的物理通道。仅当 stat_rx_synced[PCS_LANES-1:0] 的对应位为 1 时, 该总线才有效。这些输出对电平敏感。
stat_rx_vl_number_1	2	输出	该总线的值指示 PCS 通道 1 上出现的物理通道。
stat_rx_vl_number_2	2	输出	该总线的值指示 PCS 通道 2 上出现的物理通道。
stat_rx_vl_number_3	2	输出	该总线的值指示 PCS 通道 3 上出现的物理通道。

表 293: RX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_rx_synced_*	4	输出	字边界已同步。这些信号用于指示 PCS 通道是否已完成字边界同步。值为 1 表示对应 PCS 通道已实现字边界同步，并且已接收到 PCS 通道标记。 对应于第 82.3 条中所定义的 MDIO 寄存器位 3.52.7:0 和 3.53.11:0。此输出对电平敏感。
stat_rx_synced_err_*	4	输出	字边界同步错误。这些信号用于指示在相应的 PCS 通道内进行字边界同步期间，是否发生错误。值为 1 表示由于同步报头成帧位错误或者从未收到 PCS 通道标记，对应 PCS 通道已丢失字边界同步。此输出对电平敏感。
stat_rx_mf_len_err_*	4	输出	PCS 通道标记长度错误。这些信号用于指示相应通道内是否发生了 PCS 通道标记不匹配（即，接收到的 PCS 通道标记间隔字数并非 <code>ctl_rx_vl_length_minus1</code> ）。值为 1 表示对应通道接收 PCS 通道标记的间隔错误。它将保持高电平，直至消除错误状况为止。
stat_rx_mf_repeat_err_*	4	输出	PCS 通道标记连续错误。这些信号用于指示相应通道内是否发生了 4 个连续 PCS 通道标记错误。值为 1 表示对应通道内存在错误。此输出将保持高电平 (High)，直至消除错误状况为止。
stat_rx_mf_err_*	4	输出	PCS 通道标记字错误。这些信号用于表示在相应通道内检测到构成的 PCS 通道标记字错误。值为 1 表示发生了错误。此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_misaligned_*	1	输出	对齐错误。此信号表示通道对齐器并未在所有通道上都接收到期望的 PCS 通道标记。在所有通道上都至少接收到 1 个 PCS 通道标记并且至少接收到 1 个错误的通道标记后，此信号才会断言有效。此状况的发生比错误晚 1 个元帧。 如果从未正确接收到任何通道标记，则此信号不会断言有效。通道标记错误通过对应的 <code>stat_rx_mf_err</code> 信号来指示。 此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。
stat_rx_aligned_err_*	1	输出	通道对齐/去歪斜丢失。此信号表示 PCS 通道对齐期间发生错误或者 PCS 通道对齐已丢失。值为 1 表示发生了错误。此输出对电平敏感。
stat_rx_bip_err_0_*	1	输出	BIP8 错误指示符（对应 PCS 通道 0）。非 0 值表示 BIP8 签名出错。非 0 值将脉冲 1 个时钟周期。 此输出将脉冲 1 个时钟周期，以指示错误状况。
stat_rx_bip_err_1_*	1	输出	BIP8 错误指示符（对应 PCS 通道 1）。
stat_rx_bip_err_2_*	1	输出	BIP8 错误指示符（对应 PCS 通道 2）。
stat_rx_bip_err_3_*	1	输出	BIP8 错误指示符（对应 PCS 通道 3）。
stat_rx_aligned_*	1	输出	所有 PCS 通道均已对齐/去歪斜。此信号表示是否所有 PCS 通道均已对齐并去歪斜。值为 1 表示所有 PCS 通道均已对齐并去歪斜。当此信号为 1 时，表示 RX 路径已对齐并且可接收包数据。当此信号为 0 时，表示存在本地故障状况。这也对应于第 82.3 条中所定义的 MDIO 寄存器位 3.50.12。此输出对电平敏感。
stat_rx_hi_ber_*	1	输出	误码率 (BER) 过高指示符。设置为 1 时，BER 过高（根据 IEEE 802.3-2015 标准的定义）。对应于第 82.3 条中所定义的 MDIO 寄存器位 3.32.1。 此输出对电平敏感。
stat_rx_status_*	1	输出	PCS 状态。值为 1 表示 PCS 已对齐且未处于 <code>hi_ber</code> 状态。 对应于第 82.3 条中所定义的管理数据输入/输出 (MDIO) 寄存器位 3.32.12。此输出对电平敏感。
stat_rx_bad_code_*	2	输出	此信号为递增值，对应 64B/66B 编码违例数。此信号用于指示 RX PCS 接收状态机处于 IEEE 802.3-2015 标准所指定的 <code>RX_E</code> 状态。此输出可用于根据第 82.3 条的定义生成 MDIO 寄存器 3.33:7:0。
stat_rx_bad_code_valid_*	1	输出	指示 <code>stat_rx_bad_code</code> 何时有效。 当核类型为 Ethernet PCS/PMA 时，此端口可用。

表 293: RX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_rx_error_valid_*	1	输出	指示 stat_rx_error 何时有效。 当核类型为 Ethernet PCS/PMA 时, 此端口可用。
stat_rx_error_*	8	输出	测试模式不匹配数递增值。任一周期内的非 0 值均表示 RX 核中在测试模式下发生了不匹配。仅当 ctl_rx_test_pattern 设置为 1 时, 此输出才有效。此输出将脉冲 1 个时钟周期。 当核类型为 Ethernet PCS/PMA 时, 此端口可用。
stat_rx_fifo_error_*	1	输出	指示 RX FIFO 何时进入下溢或上溢状态。 如果在任意时钟周期内此输出采样为 1, 那么对应端口必须复位才能恢复正常运行。 当核类型为 Ethernet PCS/PMA 时, 此端口可用。
stat_rx_total_packets_*	2	输出	此信号为递增值, 对应已收到的数据包总数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_total_good_packets_*	1	输出	此信号为递增值, 对应已收到的有效数据包总数。仅当完全收到不含错误的数据包时, 该值才为非 0 值。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_total_bytes_*	6	输出	此信号为递增值, 对应已收到的字节总数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_total_good_bytes_*	14	输出	此信号为递增值, 对应已收到的有效字节总数。仅当完全收到不含错误的数据包时, 该值才为非 0 值。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_packet_small_*	2	输出	此信号为递增值, 对应所含数据小于 64 字节的所有包数。小于 16 字节的包将被丢弃。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_jabber_*	1	输出	此信号为递增值, 对应含无效 FCS 且长度超过 ctl_rx_max_packet_len 的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_packet_large_*	1	输出	此信号为递增值, 对应所含数据超过 9,215 字节的所有包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_oversize_*	1	输出	此信号为递增值, 对应含有效 FCS 且长度超过 ctl_rx_max_packet_len 的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_undersize_*	2	输出	此信号为递增值, 对应含有效 FCS 且长度小于 stat_rx_min_packet_len 的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_toolong_*	1	输出	此信号为递增值, 对应含有效和无效 FCS 且长度超过 ctl_rx_max_packet_len 的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_fragment_*	2	输出	此信号为递增值, 对应含无效 FCS 且长度小于 stat_rx_min_packet_len 的数据包数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_packet_64_bytes_*	1	输出	此信号为递增值, 对应已收到的有效和无效数据包 (含 64 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_packet_65_127_bytes_*	1	输出	此信号为递增值, 对应已收到的有效和无效数据包 (含 65 到 127 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_packet_128_255_bytes_*	1	输出	此信号为递增值, 对应已收到的有效和无效数据包 (含 128 到 255 字节) 数。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。

表 293：RX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_rx_packet_256_511_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 256 到 511 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_512_1023_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 512 到 1,023 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_1024_1518_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 1,024 到 1,518 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_1519_1522_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 1,519 到 1,522 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_1523_1548_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 1,523 到 1,548 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_1549_2047_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 1,549 到 2,047 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_2048_4095_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 2,048 到 4,095 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_4096_8191_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 4,096 到 8,191 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_8192_9215_bytes_*	1	输出	此信号为递增值，对应已收到的有效和无效数据包（含 8,192 到 9,215 字节）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_bad_fcs_*	2	输出	无效 FCS 指示符。该总线上的值用于指示接收到的包带有无效 FCS，但不带有含 stomp 标记的 FCS。含 stomp 标记的 FCS 定义为期望的有效 FCS 的按位反转。此输出将脉冲 1 个时钟周期，以指示错误状况。脉冲可在不间断的连续周期内发生。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_packet_bad_fcs_*	1	输出	此信号为递增值，对应具有 FCS 错误的数据包（所含字节数为 64 到 ctl_rx_max_packet_len 之间）数。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_stomped_fcs_*	2	输出	含 stomp 标记的 FCS 指示符。该总线上的值用于指示接收到的包带有含 stomp 标记的 FCS。含 stomp 标记的 FCS 定义为期望的有效 FCS 的按位反转。此输出将脉冲 1 个时钟周期，以指示含 stomp 标记。脉冲可在不间断的连续周期内发生。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。
stat_rx_bad_preamble_*	1	输出	对应无效前导码的递增值。此信号用于指示接收到以太网数据包之前是否已接收到有效的前导码。值为 1 表示接收到的前导码无效。 注释： 检测到无效的前导码时，stat_rx_bad_preamble 信号即断言有效，与 ctl_rx_check_preamble 信号的设置无关。 当核类型为 Ethernet MAC+PCS/PMA 时，此端口可用。

表 293: RX 路径控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_rx_bad_preamble_*	1	输出	对应无效 SFD 的递增值。此信号用于指示接收到以太网数据包之前是否已接收到有效的 SFD。值为 1 表示接收到的 SFD 无效。 注释: 检测到无效的 SFD 时, stat_rx_bad_preamble 信号即断言有效, 与 ctl_rx_check_preamble 信号的设置无关。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_got_signal_os_*	1	输出	Signal OS 指示。如果该位采样为 1, 则表示接收到“Signal OS”字。 注释: 在以太网网络中不应接收到 Signal OS。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_test_pattern_mismatch_*	2	输出	测试模式不匹配数递增值。任一周期内的非 0 值均表示 RX 核中在测试模式下发生的不匹配次数。仅当 ctl_rx_test_pattern 设置为 1 时, 此输出才有效。此输出可用于根据第 82.3 条的定义生成 MDIO 寄存器 3.43.15:0。此输出将脉冲 1 个时钟周期。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_truncated_*	1	输出	包截位指示符。值为 1 表示当前进行中的包因长度超过 ctl_rx_max_packet_len[14:0] 而被截位。此输出将脉冲 1 个时钟周期, 以指示截位状况。脉冲可在不间断的连续周期内发生。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_local_fault_*	1	输出	当 stat_rx_internal_local_fault 或 stat_rx_received_local_fault 断言有效时, 此输出为高电平 (High)。此输出对电平敏感。
stat_rx_remote_fault_*	1	输出	远端故障指示状态。如果该位采样为 1, 则表示检测到远端故障状况。如果该位采样为 0, 则表示不存在远端故障状况。此输出对电平敏感。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_internal_local_fault_*	1	输出	当由于以下任一操作导致生成内部本地故障时, 此信号将转为高电平 (High): 测试模式生成、通道对齐错误或误码率过高。只要故障状况仍然存在, 此信号就会保持高电平 (High)。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。
stat_rx_received_local_fault_*	1	输出	当接收到来自链路伙伴的本地故障字数足以触发 IEEE 故障状态机所指定的故障条件时, 此信号将转为高电平 (High)。只要故障状况仍然存在, 此信号就会保持高电平 (High)。 当核类型为 Ethernet MAC+PCS/PMA 时, 此端口可用。

TX 暂停接口控制信号、状态信号和统计数据信号

在“MAC Options” (MAC 选项) 选项卡下选中“Enable TX Flow Control Logic” (启用 TX 流量控制逻辑), 并且核类型为“Ethernet MAC+PCS/PMA”时, 本节下所述端口可用。

表 294: TX 暂停接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_tx_pause_req_*	9	输入	如果该总线的某个位设置为 1, 那么核会使用 ctl_tx_pause_quanta[8:0][15:0] 总线上关联的暂停量值来发射暂停包。如果 bit[8] 设置为 1, 那么将发射全局暂停包。所有其它位设置为 1 都会导致发射优先暂停包。
ctl_tx_pause_enable_*	9	输入	TX 暂停使能信号。此输入用于启用对应优先级的暂停量的处理。此信号用于对暂停包的发射进行门控。 未选中 AXI4-Lite 接口时, 此端口可用。

表 294：TX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_tx_resend_pause_*	1	输入	重新发射挂起的暂停包。当此输入采样为 1 时，所有挂起的暂停包都将尽快（即，在当前进行中的包完成后）重新发射，并且重新发射计数器将复位。此输入应脉冲至 1，并且每次保持 1 个周期不变。
ctl_tx_pause_quanta0	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta1	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta2	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta3	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta4	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta5	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta6	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta7	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_quanta8	16	输入	这些总线用于指示基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级要发射的暂停量。ctl_tx_pause_quanta[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer0	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer1	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer2	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。未选中 AXI4-Lite 接口时，此端口可用。

表 294：TX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_tx_pause_refresh_timer3	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer4	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer5	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer6	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer7	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_pause_refresh_timer8	16	输入	此总线用于设置基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中每个优先级，暂停包的重新发射时间。 ctl_tx_pause_refresh_timer[8] 的值用于全局暂停操作。所有其它值均用于优先暂停操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_da_gpp_*	48	输入	发射全局暂停包的目标地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_sa_gpp_*	48	输入	发射全局暂停包的源地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_ethertype_gpp_*	16	输入	发射全局暂停包的以太类型 (Ethertype)。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_opcode_gpp_*	16	输入	发射全局暂停包的操作代码。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_da_ppp_*	48	输入	发射优先暂停包的目标地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_sa_ppp_*	48	输入	发射优先暂停包的源地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_ethertype_ppp_*	16	输入	发射优先暂停包的以太类型 (Ethertype)。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_tx_opcode_ppp_*	16	输入	发射优先暂停包的操作代码。 未选中 AXI4-Lite 接口时，此端口可用。
stat_tx_pause_valid_*	9	输出	如果该总线的某个位设置为 1，则表示 40G/50G High Speed Ethernet Subsystem 已发射暂停包。如果 bit[8] 设置为 1，则表示已发射全局暂停包。所有其它位设置为 1 都会导致发射优先暂停包。
stat_tx_unicast_*	1	输出	此信号为递增值，对应有有效单播数据包数。
stat_tx_multicast_*	1	输出	此信号为递增值，对应有有效多播数据包数。

表 294：TX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_tx_broadcast_*	1	输出	此信号为递增值，对应有有效广播数据包数。
stat_tx_vlan_*	1	输出	此信号为递增值，对应含有效 802.1Q 标签的 VLAN 数据包数。
stat_tx_pause_*	1	输出	此信号为递增值，对应含有效 FCS 的 802.3x Ethernet MAC 暂停包数。
stat_tx_user_pause_*	1	输出	此信号为递增值，对应含有效 FCS 的基于优先级的暂停包数。

RX 暂停接口控制信号、状态信号和统计数据信号

在“MAC Options”（MAC 选项）选项卡下选中“Enable RX Flow Control Logic”（启用 RX 流量控制逻辑），并且核类型为“Ethernet MAC+PCS/PMA”时，本节下所述端口可用。

表 295：RX 暂停接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_rx_forward_control_*	1	输入	值为 1 表示该核会将控制包转发给您。值为 0 会导致该核删除控制包。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_pause_ack_*	9	输入	暂停确认。该总线用于确认收到来自用户逻辑的暂停帧。
ctl_rx_check_ack_*	1	输入	等待确认。如果此输入设置为 1，那么该核会使用 ctl_rx_pause_ack[8:0] 总线进行暂停处理。如果此输入设置为 0，则不使用 ctl_rx_pause_ack[8:0]。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_pause_enable_*	9	输入	RX 暂停启用。此输入用于启用对应优先级的暂停量的处理。 注释： 此信号仅影响 RX 用户接口，不影响暂停处理逻辑。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_enable_gcp_*	1	输入	值为 1 即表示启用全局控制包处理。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_check_mcast_gcp_*	1	输入	值为 1 即表示启用全局控制多播目标地址处理。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_check_ucast_gcp_*	1	输入	值为 1 即表示启用全局控制单播目标地址处理。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_pause_da_ucast_*	48	输入	暂停处理的单播目标地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_check_sa_gcp_*	1	输入	值为 1 即表示启用全局控制源地址处理。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_pause_sa_*	48	输入	暂停处理的源地址。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_check_etype_gcp_*	1	输入	值为 1 即表示启用全局控制以太类型 (Ethertype) 处理。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_etype_gcp_*	16	输入	全局控制处理的以太类型 (Ethertype) 字段。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_rx_check_opcode_gcp_*	1	输入	值为 1 即表示启用全局控制操作代码 (opcode) 处理。 未选中 AXI4-Lite 接口时，此端口可用。

表 295: RX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_rx_opcode_min_gcp_*	16	输入	最小全局控制操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_opcode_max_gcp_*	16	输入	最大全局控制操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_enable_pcp_*	1	输入	值为 1 即表示启用优先控制包处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_mcast_pcp_*	1	输入	值为 1 即表示启用优先控制多播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_ucast_pcp_*	1	输入	值为 1 即表示启用优先控制单播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_pause_da_mcast_*	48	输入	暂停处理的多播目标地址。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_sa_pcp_*	1	输入	值为 1 即表示启用优先控制源地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_etype_pcp_*	1	输入	值为 1 即表示启用优先控制以太类型 (Ethertype) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_etype_pcp_*	16	输入	优先控制处理的以太类型 (Ethertype) 字段。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_opcode_pcp_*	1	输入	值为 1 即表示启用优先控制操作代码 (opcode) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_opcode_min_pcp_*	16	输入	最小优先级控制操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_opcode_max_pcp_*	16	输入	最大优先级控制操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_enable_gpp_*	1	输入	值为 1 即表示启用全局暂停包处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_mcast_gpp_*	1	输入	值为 1 即表示启用全局暂停多播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_ucast_gpp_*	1	输入	值为 1 即表示启用全局暂停单播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_sa_gpp_*	1	输入	值为 1 即表示启用全局暂停源地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_etype_gpp_*	1	输入	值为 1 即表示启用全局暂停以太类型 (Ethertype) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_etype_gpp_*	16	输入	全局暂停处理的以太类型 (Ethertype) 字段。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_opcode_gpp_*	1	输入	值为 1 即表示启用全局暂停操作代码 (opcode) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_opcode_gpp_*	16	输入	全局暂停操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_enable_ppp_*	1	输入	值为 1 即表示启用优先暂停包处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_mcast_ppp_*	1	输入	值为 1 即表示启用优先暂停多播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。

表 295: RX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_rx_check_ucast_ppp_*	1	输入	值为 1 即表示启用优先暂停单播目标地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_sa_ppp_*	1	输入	值为 1 即表示启用优先暂停源地址处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_etype_ppp_*	1	输入	值为 1 即表示启用优先暂停以太类型 (Ethertype) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_etype_ppp_*	16	输入	优先暂停处理的以太类型 (Ethertype) 字段。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_check_opcode_ppp_*	1	输入	值为 1 即表示启用优先暂停操作代码 (opcode) 处理。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_rx_opcode_ppp_*	16	输入	优先暂停操作代码值。 未选中 AXI4-Lite 接口时, 此端口可用。
stat_rx_unicast_*	1	输出	此信号为递增值, 对应有效单播数据包数。
stat_rx_multicast_*	1	输出	此信号为递增值, 对应有效多播数据包数。
stat_rx_broadcast_*	1	输出	此信号为递增值, 对应有效广播数据包数。
stat_rx_vlan_*	1	输出	此信号为递增值, 对应含有效 802.1Q 标签的 VLAN 数据包数。
stat_rx_pause_*	1	输出	此信号为递增值, 对应含有效 FCS 的 802.3x Ethernet MAC 暂停包数。
stat_rx_user_pause_*	1	输出	此信号为递增值, 对应含有效 FCS 的基于优先级的暂停包数。
stat_rx_inrangeerr_*	1	输出	此信号为递增值, 对应含 “Length” (长度) 字段错误但具有有效 FCS 的数据包数。
stat_rx_pause_valid_*	9	输出	该总线用于指示已收到暂停包, 并且 stat_rx_pause_quanta[8:0] [15:0] 总线上的关联暂停量有效, 且必须将其用于暂停处理。如果收到的是 802.3x Ethernet MAC 暂停包, 则 bit[8] 设置为 1。
stat_rx_pause_quanta0	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta1	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta2	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta3	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta4	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta5	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta6	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。
stat_rx_pause_quanta7	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中, 针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包, 那么此暂停量将置于值 [8] 内。

表 295: RX 暂停接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_rx_pause_quanta8	16	输出	这些总线用于指示在基于优先级的暂停操作和全局暂停操作中，针对 8 个优先级中的每个优先级所接收到的暂停量。如果收到的是 802.3x Ethernet MAC 暂停包，那么此暂停量将置于值 [8] 内。
stat_rx_pause_req_*	9	输出	暂停请求信号。当 RX 收到有效的暂停帧时，它会将该总线的对应位设置为 1，其值保持为 1 不变，直至完成暂停包的处理为止。

IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号

在“MAC Options”选项卡下选中“Enable_Time_Stamping”时，本节下所述端口可用。

表 296: IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_tx_systemtimerin_*	80	输入	适用于 TX 的系统定时器输入。 在正常时钟模式下，时间格式遵循 IEEE 1588 格式，以 48 位表示秒，32 位表示纳秒。 在透明时钟模式下，位 63 预期为零，位 62:16 的单位为纳秒，位 15:0 的单位为小数纳秒。请参阅 IEEE 1588v2 以了解相关表示法的定义。 此输入必须位于 TX 时钟域内。
ctl_rx_systemtimerin_*	80	输入	适用于 RX 的系统定时器输入。 在正常时钟模式下，时间格式遵循 IEEE 1588 格式，以 48 位表示秒，32 位表示纳秒。 在透明时钟模式下，位 63 预期为零，位 62:16 的单位为纳秒，位 15:0 的单位为小数纳秒。请参阅 IEEE 1588v2 以了解相关表示法的定义。 此输入必须与通道 0 RX SerDes 位于相同时钟域内。
ctl_tx_ptp_1step_enable_*	1	输入	设为 1 时，该位即启用单步操作。 在“Configuration”选项卡下未选中“Include AXI4-Lite”时，此端口可用。
ctl_tx_ptp_latency_adjust_*	11	输入	该总线可用于调整与双步时间戳有关的单步 TX 时间戳。位 [10:3] 的单位为纳秒，而位 [2:0] 的单位则为小数纳秒。 在“Configuration”选项卡下未选中“Include AXI4-Lite”时，此端口可用。
ctl_tx_ptp_vlane_adjust_mode_*	1	输入	设置 vlan 调整模式。 在“Configuration”选项卡下未选中“Include AXI4-Lite”时，此端口可用。
ctl_ptp_transpclk_mode_*	1	输入	设为 1 时，此输入会将时间戳逻辑置于透明时钟模式下。在此模式下，系统定时器输入将解释为纠正值。TX 将根据 IEEE 1588v2 中定义的进程，把该纠正值添加到 TX 时间戳。纠正值的符号位假定为 0（正时间）。 这样对应的传入 PTP 包纠正字段应已按适当的 RX 时间戳完成调整。 在“Configuration”选项卡下未选中“Include AXI4-Lite”时，此端口可用。
stat_tx_ptp_fifo_read_error_*	1	输出	发射 PTP FIFO 写入错误。此状态值为 1 表示 PTP Tag 写入期间发生错误。需 TX 路径复位才能清除此错误。
stat_tx_ptp_fifo_write_error_*	1	输出	发射 PTP FIFO 读取错误。此状态值为 1 表示 PTP Tag 读取期间发生错误。需 TX 路径复位才能清除此错误。

表 296: IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
tx_ptp_1588op_in_*	2	输入	2'b00 - No operation: 表示无操作, 不记录时间戳, 并且不修改帧。 2'b01 - 1-step: 表示单步, 应记录时间戳并将其插入帧。 2'b10 - 2-step: 表示双步, 应记录时间戳并使用对应双步操作的额外端口将该时间戳返回到客户端。不修改帧本身。 2'b11 - Reserved: 表示保留, 相当于“无操作”。
tx_ptp_tag_field_in_*	16	输入	该字段的使用取决于 1588 操作。
tx_ptp_tstamp_valid_out_*	1	输出	此位用于表示 TX 上当前呈现的有效时间戳。
tx_ptp_tstamp_tag_out_*	16	输出	对应于 tx_ptp_tag_field_in[15:0] 的标签输出
tx_ptp_tstamp_out_*	80	输出	已发射的包 SOP 的时间戳, 对应于它通过捕获平面的时间。用于双步 1588 操作。 该总线中包含的位的表示法与定时器输入相同。
rx_ptp_tstamp_out_*	80	输出	已接收的包 SOP 的时间戳, 对应于它通过捕获平面的时间。请注意, 只要针对一个分段断言 SOP 有效, 则从断言有效的时钟周期开始, 此信号即生效。 该总线中包含的位的表示法与定时器输入相同。
tx_ptp_upd_chksum_in_*	1	输入	TX 已更新 UPD 校验和值。 在“MAC options”选项卡下针对“IEEE PTP Operation Mode”选中“One Step”时, 此端口可用。
tx_ptp_tstamp_offset_in_*	16	输入	TX PTP 时间戳偏移。 在“MAC options”选项卡下针对“IEEE PTP Operation Mode”选中“One Step”时, 此端口可用。
tx_ptp_chksum_offset_in_*	16	输入	TX PTP 校验和偏移。 在“MAC options”选项卡下针对“IEEE PTP Operation Mode”选中“One Step”时, 此端口可用。
tx_ptp_pcs_lane_out_*	2	输出	该总线用于指示检测到对应时间戳的 SOP 所在的 PCS 通道。 请注意, 只要针对一个分段断言 SOP 有效, 则从断言有效的时钟周期开始, 此信号即生效。
rx_ptp_pcs_lane_out_*	2	输出	该总线用于指示检测到对应时间戳的 SOP 所在的 PCS 通道。 请注意, 只要针对一个分段断言 SOP 有效, 则从断言有效的时钟周期开始, 此信号即生效。
rx_lane_aligner_fill_0	7	输出	此输出用于指示 PCS lane0 的对齐缓冲器的填充级别。此信息可供 PTP 应用配合 rx_ptp_pcs_lane_out_* 信号一起使用, 以便调整即将到达的 SOP 的通道偏差。单位为 SerDes 时钟周期数。
rx_lane_aligner_fill_1	7	输出	此输出用于指示 PCS lane1 的对齐缓冲器的填充级别。
rx_lane_aligner_fill_2	7	输出	此输出用于指示 PCS lane2 的对齐缓冲器的填充级别。
rx_lane_aligner_fill_3	7	输出	此输出用于指示 PCS lane3 的对齐缓冲器的填充级别。
gtm_txusrclk2_*	1	输入	TX 时钟输入, 连接到核。 注释: 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并且 GT 类型为 GTM 时, 此端口可用。
gtm_rxusrclk2_*	1	输入	RX 时钟输入, 连接到核。 注释: 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项并且 GT 类型为 GTM 时, 此端口可用。
gtm_txprgdivresetdone_*	1	输入	如需了解更多信息, 请参阅《Virtex UltraScale+ FPGA GTM Transceivers Wizard LogiCORE IP 产品指南》(PG315)。

表 296: IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
gtm_rxprgdivresetdone_*	1	输入	如需了解更多信息, 请参阅《Virtex UltraScale+ FPGA GTM Transceivers Wizard LogiCORE IP 产品指南》(PG315)。

AN 和 LT 接口控制信号、状态信号和统计数据信号

在“Configuration”选项卡下选中“Include AN/LT Logic”时, 本节下所述端口可用。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
an_clk_*	1	输入	自动协商电路的输入时钟。 此信号应为自由运行的时钟。
an_reset_*	1	输入	与 an_clk 域对应的异步高电平有效复位。
an_loc_np_data_*	48	输入	本地下一页代码字。这是设置 loc_np 输入的情况下所使用的 48 位代码字。在此数据字段中, 位 NP、ACK 和 T 以及位元位置 15、14、12 和 11 将不会随下一页代码字一起传输。这些位是在 AN IP 中生成的。但位元位置 13 处的消息协议位 MP 将进行传输。
an_lp_np_data_*	48	输出	链路伙伴下一页数据信号。此 48 位代码字由 AN IP 驱动, 包含来自远端链路伙伴的 48 位下一页代码字。
lt_tx_sof_*	4	输出	这是链路训练信号, 在每个训练帧开始后的 1 个 tx_serdes_clk 周期内, 此信号将断言有效。此信号是为需要计算训练帧数量或者需要将事件同步至训练帧的输出的应用而提供的。
ctl_autoneg_enable_*	1	输入	自动协商的使能信号。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_autoneg_bypass_*	1	输入	输入信号, 用于禁用自动协商并绕过自动协商功能。如果断言此输入有效, 将关闭自动协商, 但 PCS 将连接到输出以允许执行操作。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_nonce_seed_*	8	输入	8 位种子, 用于初始化随机数 (nonce) 字段多项式生成器。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_pseudo_sel_*	1	输入	选择位 49 随机位生成器的多项式生成器。如果此输入为 1, 则多项式为 x^7+x^6+1 。如果此输入为 0, 则多项式为 x^7+x^3+1 。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_restart_negotiation_*	1	输入	此输入用于触发重新启动自动协商, 与当前电路状态无关。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_local_fault_*	1	输入	此输入信号用于设置发射链路代码字的 remote_fault 位。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_pause_*	1	输入	此输入信号用于设置发射链路代码字的 PAUSE 位 (C0)。如果该核不支持暂停, 则此信号可能不存在。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_asmdir_*	1	输入	此输入信号用于设置发射链路代码字的 ASMDIR 位 (C1)。如果该核不支持暂停, 则此信号可能不存在。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_fec_10g_request_*	1	输入	此信号用于告知链路伙伴, 本地工作站正在请求在 10Gb/s 通道协议上使用第 74 条 FEC。 未选中 AXI4-Lite 接口时, 此端口可用。
ctl_an_fec_25g_rs_request_*	1	输入	此信号用于告知链路伙伴, 本地工作站正在请求在 25Gb/s 通道协议上使用 rs FEC (第 91 或 108 条)。 未选中 AXI4-Lite 接口时, 此端口可用。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_an_fec_25g_baser_request_*	1	输入	此信号用于告知链路伙伴，本地工作站正在请求在 25Gb/s 通道协议上使用第 74 条 FEC。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_an_fec_ability_override_*	1	输入	用于在发射链路代码字中设置第 74 条 FEC 功能位。如果设置此输入，则清除发射链路代码字中的 FEC 功能位。如果该 IP 核不支持第 74 条 FEC，则此信号可能不存在。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_an_loc_np_*	1	输入	本地下一页 (Local Next Page) 信号指示符。如果此位为 1，则 AN IP 会将 loc_np_data 输入处的下一页代码字传输到远程链路伙伴。如果此位为 0，则 AN IP 不会启动下一页协议。如果链路伙伴包含多个待发送的下一页信号，并且“loc_np”位已清零，那么 AN IP 将传输空消息页面。
ctl_an_ip_np_ack_*	1	输入	链路伙伴下一页确认信号。此信号用于告知 AN IP，本地主机已读取位于 lp_np_data 输出管脚处的下一页数据（来自远端链路伙伴）。当此信号变为高电平时，表示 AN IP 确认已接收发送到远端链路伙伴的下一页代码字，并启动下一个代码字的传输。在此期间，AN IP 会移除 lp_np 信号，直至新的下一页信息可用为止。
ctl_an_cl91_fec_request_*	1	输入	该位用于请求第 91 条 FEC。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_an_cl91_fec_ability_*	1	输入	该位用于设置第 91 条 FEC 功能。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_an_ability_1000base_kx_*	1	输入	这些输入用于识别发射链路代码字中播发到链路伙伴的以太网协议功能。值为 1 表示接口支持此协议并播发此信息。
ctl_an_ability_10gbase_kx4_*	1	输入	
ctl_an_ability_10gbase_kr_*	1	输入	
ctl_an_ability_40gbase_kr4_*	1	输入	
ctl_an_ability_40gbase_cr4_*	1	输入	
ctl_an_ability_100gbase_cr10_*	1	输入	
ctl_an_ability_100gbase_kp4_*	1	输入	
ctl_an_ability_100gbase_kr4_*	1	输入	
ctl_an_ability_100gbase_cr4_*	1	输入	
ctl_an_ability_25gbase_krcr_s_*	1	输入	
ctl_an_ability_25gbase_krcr_*	1	输入	
ctl_an_ability_25gbase_kr1_*	1	输入	
ctl_an_ability_25gbase_cr1_*	1	输入	
ctl_an_ability_50gbase_kr2_*	1	输入	
ctl_an_ability_50gbase_cr2_*	1	输入	
ctl_lt_training_enable_*	1	输入	启用链路训练。禁用链路训练时，所有 PCS 通道都以任务模式进行操作。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_restart_training_*	1	输入	此信号用于触发重新启动链路训练，与当前状态无关。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_rx_trained_*	4	输入	此信号断言有效即可指示接收器 FIR 滤波器系数已全部设置完成，并且训练的接收器部分已完成。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_preset_to_tx_*	4	输入	此信号用于设置预置位的值，该位用于发射到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_lt_initialize_to_tx_*	4	输入	此信号用于设置初始位的值，该位用于发射到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_pseudo_seed0_*	11	输入	此 11 位信号用于植入训练模式生成器的种子。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_p1_to_tx0_*	2	输入	此 2 位字段用于设置 k+1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k0_to_tx0_*	2	输入	此 2 位字段用于设置 k0 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_m1_to_tx0_*	2	输入	此 2 位字段用于设置 k-1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_p1_to_tx0_*	2	输入	此 2 位字段用于设置 k+1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat0_to_tx0_*	2	输入	此 2 位字段用于设置 k0 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_m1_to_tx0_*	2	输入	此 2 位字段用于设置 k-1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_pseudo_seed1_*	11	输入	此 11 位信号用于植入训练模式生成器的种子。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_p1_to_tx1_*	2	输入	此 2 位字段用于设置 k+1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k0_to_tx1_*	2	输入	此 2 位字段用于设置 k0 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_m1_to_tx1_*	2	输入	此 2 位字段用于设置 k-1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_p1_to_tx1_*	2	输入	此 2 位字段用于设置 k+1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat0_to_tx1_*	2	输入	此 2 位字段用于设置 k0 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_m1_to_tx1_*	2	输入	此 2 位字段用于设置 k-1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_pseudo_seed2_*	11	输入	此 11 位信号用于植入训练模式生成器的种子。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_p1_to_tx2_*	2	输入	此 2 位字段用于设置 k+1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_lt_k0_to_tx2_*	2	输入	此 2 位字段用于设置 k0 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_m1_to_tx2_*	2	输入	此 2 位字段用于设置 k-1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_p1_to_tx2_*	2	输入	此 2 位字段用于设置 k+1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat0_to_tx2_*	2	输入	此 2 位字段用于设置 k0 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_m1_to_tx2_*	2	输入	此 2 位字段用于设置 k-1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_pseudo_seed3_*	11	输入	此 11 位信号用于植入训练模式生成器的种子。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_p1_to_tx3_*	2	输入	此 2 位字段用于设置 k+1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k0_to_tx3_*	2	输入	此 2 位字段用于设置 k0 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_k_m1_to_tx3_*	2	输入	此 2 位字段用于设置 k-1 系数更新字段的值，该字段用于发送到训练帧的控制块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_p1_to_tx3_*	2	输入	此 2 位字段用于设置 k+1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat0_to_tx3_*	2	输入	此 2 位字段用于设置 k0 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。
ctl_lt_stat_m1_to_tx3_*	2	输入	此 2 位字段用于设置 k-1 系数更新状态的值，该状态用于发送到训练帧的状态块中的链路伙伴。 未选中 AXI4-Lite 接口时，此端口可用。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_an_link_cntl_1000base_kx_*	2	输出	来自各种以太网协议的自动协商控制器的链路控制输出。设置如下： 00: DISABLE; PCS 断开连接 01: SCAN_FOR_CARRIER; RX 已连接到 PCS 10: 不使用 11: ENABLE; PCS 已连接，可执行任务模式
stat_an_link_cntl_10gbase_kx4_*	2	输出	
stat_an_link_cntl_10gbase_kr_*	2	输出	
stat_an_link_cntl_40gbase_kr4_*	2	输出	
stat_an_link_cntl_40gbase_cr4_*	2	输出	
stat_an_link_cntl_100gbase_cr10_*	2	输出	
stat_an_link_cntl_100gbase_kp4_*	2	输出	
stat_an_link_cntl_100gbase_kr4_*	2	输出	
stat_an_link_cntl_100gbase_cr4_*	2	输出	
stat_an_link_cntl_25gbase_krcr_s_*	2	输出	
stat_an_link_cntl_25gbase_krcr_*	2	输出	
stat_an_link_cntl_25gbase_kr1_*	2	输出	
stat_an_link_cntl_25gbase_cr1_*			
stat_an_link_cntl_50gbase_kr2_*			
stat_an_link_cntl_50gbase_cr2_*			
stat_an_fec_enable_*	1	输出	用于支持在链路上使用第 74 条 FEC。
stat_an_tx_pause_enable_*	1	输出	用于支持在发射路径中进行站对站（全局）暂停包生成，以控制接收路径中的数据流。
stat_an_rx_pause_enable_*	1	输出	用于支持在接收路径中进行站对站（全局）暂停包解释，以控制来自发射器的数据流。
stat_an_autoneg_complete_*	1	输出	指示自动协商已完成，并且已接收到来自 PCS 的 rx 链路状态。
stat_an_parallel_detection_fault_*	1	输出	指示自动协商期间发生并行检测故障。
stat_an_start_tx_disable_*	1	输出	此信号在自动协商期间的 断言有效并保持 1 个 AN_CLK 周期，以在自动协商开始时作为 TX_DISABLE 阶段的起始信号。
stat_an_start_an_good_check_*	1	输出	此信号在自动协商期间的 断言有效并保持 1 个 AN_CLK 周期，以作为 AN_GOOD_CHECK 阶段的起始信号，前提是所选协议已启用并且电路正在等待 rx_pcs_status。
stat_an_lp_ability_1000base_kx_*	1	输出	这些信号用于指示从链路伙伴播发的协议。当 stat_an_lp_ability_valid 输出信号断言有效时，这些信号全部变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_10gbase_kx4_*	1	输出	
stat_an_lp_ability_10gbase_kr_*	1	输出	
stat_an_lp_ability_40gbase_kr4_*	1	输出	
stat_an_lp_ability_40gbase_cr4_*	1	输出	
stat_an_lp_ability_100gbase_cr10_*	1	输出	
stat_an_lp_ability_100gbase_kp4_*	1	输出	
stat_an_lp_ability_100gbase_kr4_*	1	输出	
stat_an_lp_ability_100gbase_cr4_*	1	输出	
stat_an_lp_ability_25gbase_krcr_s_*	1	输出	
stat_an_lp_ability_25gbase_krcr_*	1	输出	

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_an_lp_pause_*	1	输出	此信号用于指示来自链路伙伴的接收链路代码字的 PAUSE 位 (C0) 的播发值。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_asm_dir_*	1	输出	此信号用于指示来自链路伙伴的接收链路代码字的 ASMDIR 位 (C1) 的播发值。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_rf_*	1	输出	此位用于指示链路伙伴远端故障。
stat_an_lp_fec_10g_ability_*	1	输出	此信号用于指示与链路伙伴正在播发的 10 Gb/s 通道协议相关联的第 74 条 FEC 功能。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_fec_10g_request_*	1	输出	此信号用于指示链路伙伴正在请求在 10 Gb/s 通道协议上使用第 74 条 FEC。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_fec_25g_rs_request_*	1	输出	此信号用于指示链路伙伴正在请求在 25 Gb/s 通道协议上使用第 91 条 (或第 108 条) rs FEC。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_fec_25g_baser_request_*	1	输出	此信号指示链路伙伴正在请求在 25 Gb/s 通道 base-r 协议上使用第 74 条 FEC。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_autoneg_able_*	1	输出	此输出信号用于指示链路伙伴能够执行自动协商。当 stat_an_lp_ability_valid 输出信号断言有效时, 此信号即变为有效。
stat_an_lp_ability_valid_*	1	输出	此信号用于指示何时所有链路伙伴播发将变为有效。
stat_an_loc_np_ack_*	1	输出	此信号可指示用于接收 loc_np_data 输入管脚上显示的本地下一页数据的本地主机。当 AN IP 在输入管脚 oc_np_data 上对下一页数据进行采样时, 此信号将脉冲至高电平并保持 1 个时钟周期。当本地主机检测到此信号为高电平时, 它必须将位于 loc_np_data 输入管脚处的 48 位下一页代码字替换为要发送的下一个 48 位代码字。如果本地主机没有其它下一页要发送, 则它必须将 loc_np 输入清零。
stat_an_lp_np_*	1	输出	链路伙伴下一页信号。此信号用于指示在 lp_np_data 输出管脚处存在来自远端链路伙伴的有效的 48 位下一页代码字。当 lp_np_ack 输入信号被驱动至高电平时, 此信号将被驱动至低电平, 表示本地主机已读取下一页数据。它将保持低电平, 直至在 lp_np_data 输出管脚上下一个代码字变为可用为止, 届时 lp_np 输出将被重新驱动至高电平。
stat_an_lp_ability_25gbase_kr1_*	1	输出	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时, 此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_link_cntl_25gbase_cr1_*	1	输出	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时, 此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_50gbase_kr2_*	1	输出	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时, 此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_50gbase_cr2_*	1	输出	用于指示从链路伙伴播发的协议。当 stat_an_lp_extended_ability_valid 输出信号断言有效时, 此信号即变为有效。值为 1 表示此协议已作为链路伙伴支持的协议进行播发。
stat_an_lp_ability_extended_fec_*	4	输出	此输出用于指示 FEC 扩展功能。
stat_an_rs_fec_enable_*	1	输出	用于支持在链路上使用第 91 条 FEC。
stat_an_lp_extended_ability_valid_*	1	输出	当该位为 1 时, 表示检测到的扩展功能有效。
stat_lt_signal_detect_*	4	输出	此信号用于指示相应的链路训练状态机已进入 SEND_DATA 状态, 在此状态下可恢复正常 PCS 操作。
stat_lt_training_*	4	输出	此信号用于指示何时相应的链路训练状态机正在执行链路训练。

表 297：AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_lt_training_fail_*	4	输出	此信号在链路训练期间断言有效，前提是对应的链路训练状态机在训练周期期间检测到超时。
stat_lt_rx_sof_*	4	输出	此输出将在 1 个 RX SerDes 时钟周期内保持高电平，以指示链路训练帧开始。
stat_lt_frame_lock_*	4	输出	链路训练开始后，将针对每个 PMD 通道断言这些信号有效，前提是对应的链路训练接收器能够与链路伙伴建立帧同步。
stat_lt_preset_from_rx_*	4	输出	此信号用于反映从控制块中的链路伙伴接收到的预置控制位的值。
stat_lt_initialize_from_rx_*	4	输出	此信号用于反映从控制块中的链路伙伴接收到的初始化控制位的值。
stat_lt_k_p1_from_rx0_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k+1 系数的更新控制位。
stat_lt_k0_from_rx0_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k0 系数的更新控制位。
stat_lt_k_m1_from_rx0_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k-1 系数的更新控制位。
stat_lt_stat_p1_from_rx0_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k+1 系数的更新状态位。
stat_lt_stat0_from_rx0_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k0 系数的更新状态位。
stat_lt_stat_m1_from_rx0_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k-1 系数的更新状态位。
stat_lt_k_p1_from_rx1_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k+1 系数的更新控制位。
stat_lt_k0_from_rx1_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k0 系数的更新控制位。
stat_lt_k_m1_from_rx1_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k-1 系数的更新控制位。
stat_lt_stat_p1_from_rx1_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k+1 系数的更新状态位。
stat_lt_stat0_from_rx1_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k0 系数的更新状态位。
stat_lt_stat_m1_from_rx1_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k-1 系数的更新状态位。
stat_lt_k_p1_from_rx2_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k+1 系数的更新控制位。
stat_lt_k0_from_rx2_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k0 系数的更新控制位。
stat_lt_k_m1_from_rx2_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k-1 系数的更新控制位。
stat_lt_stat_p1_from_rx2_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k+1 系数的更新状态位。
stat_lt_stat0_from_rx2_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k0 系数的更新状态位。
stat_lt_stat_m1_from_rx2_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k-1 系数的更新状态位。
stat_lt_k_p1_from_rx3_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k+1 系数的更新控制位。
stat_lt_k0_from_rx3_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k0 系数的更新控制位。

表 297: AN 和 LT 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
stat_lt_k_m1_from_rx3_*	2	输出	此 2 位字段用于指示从控制块中的链路伙伴接收到的 k-1 系数的更新控制位。
stat_lt_stat_p1_from_rx3_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k+1 系数的更新状态位。
stat_lt_stat0_from_rx3_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k0 系数的更新状态位。
stat_lt_stat_m1_from_rx3_*	2	输出	此 2 位字段用于指示从状态块中的链路伙伴接收到的 k-1 系数的更新状态位。
stat_an_rxcdrhold_*	1	输出	用于指示 rx cdr hold 信号。

第 74 条 FEC 接口控制信号、状态信号和统计数据信号

在“Configuration”（配置）选项卡下选中“Clause 74 (BASE-KR FEC)”时，本节下所述端口可用。

表 298: 第 74 条 FEC 接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_fec_tx_enable_*	1	输入	此信号断言有效即可对发射的数据启用第 74 条 FEC 编码。
ctl_fec_rx_enable_*	1	输入	此信号断言有效即可对接收的数据启用第 74 条 FEC 解码。
stat_fec_inc_correct_count_*	4	输出	此信号每隔约 32 个代码字就会断言有效，前提是同时断言 ctl_rx_fec_enable 有效，且 FEC 解码器已检测到并已纠正对应帧中的位元错误。
stat_fec_inc_cant_correct_count_*	4	输出	此信号每隔约 32 个代码字就会断言有效，前提是同时断言 ctl_rx_fec_enable 有效，且 FEC 解码器已检测到帧中的位元错误但无法将其纠正。
stat_fec_lock_error_*	4	输出	ctl_fec_rx_enable 断言有效，前提是 FEC 解码器经过约 5 ms 后仍无法检测到帧边界。检测到帧边界后，即清除此信号。
stat_fec_rx_lock_*	4	输出	此信号断言有效的前提是同时断言 ctl_fec_rx_enable 有效，且 FEC 解码器可检测到帧边界。

第 91 条 RS-FEC 接口控制信号、状态信号和统计数据信号

在“Configuration”（配置）选项卡下选中“Clause 91 (RS-FEC)”时，本节下所述端口可用。

表 299: 第 91 条 RS-FEC 接口控制信号、状态信号和统计数据信号

名称	大小	I/O	描述
ctl_rsfc_enable_*	1	输入	复位后，更改才会生效。复位后，在首个周期内将对新的值进行采样。启用 RS-FEC 功能。 1: 启用 RS-FEC 0: 绕过 RS-FEC
ctl_rx_rsfc_enable_correction_*	1	输入	复位后，更改才会生效。复位后，在首个周期内将对新的值进行采样。等效于 MDIO 寄存器 1.200.0。 0: 解码器会执行错误检测，但不执行纠错（请参阅 IEEE 802.3802.3 第 91.5.3.3 条） 1: 解码器还会执行纠错

表 299：第 91 条 RS-FEC 接口控制信号、状态信号和统计数据信号 (续)

名称	大小	I/O	描述
ctl_rx_rsfc_enable_indication_*	1	输入	复位后，更改才会生效。复位后，在首个周期内将对新的值进行采样。等效于 MDIO 寄存器 1.200.1 0：绕过错误指示功能（请参阅 IEEE 802.3 标准第 91.5.3.3 条） 1：解码器向 PCS 子层发出错误指示
ctl_rsfc_ieee_error_indication_mode_*	1	输入	复位后，更改才会生效。复位后，在首个周期内将对新的值进行采样。 1：核遵循 IEEE RS-FEC 规范 0：如果 ctl_rx_rsfc_enable_correction 和 ctl_rx_rsfc_enable_indication 均为 0，那么将绕过 RS 解码器
stat_tx_rsfc_block_lock_*	1	输出	TX PCS 块锁定状态 0：已解锁 1：已锁定
stat_tx_rsfc_lane_alignment_status_*	1	输出	TX PCS 帧对齐状态 0：未对齐 1：已对齐
stat_rx_rsfc_am_lock0	1	输出	RX 通道 0 锁定状态 0：已解锁 1：已锁定
stat_rx_rsfc_am_lock1	1	输出	RX 通道 1 锁定状态 0：已解锁 1：已锁定
stat_rx_rsfc_lane_alignment_status_*	1	输出	RX 对齐状态 0：未对齐 1：已对齐
stat_rx_rsfc_lane_fill_0	14	输出	RX 通道 0 额外延迟。 每条延迟总线的头 7 个位 [13:7] 即为因去歪斜而导致添加的额外时钟周期延迟数。每条延迟总线的末 7 个位 [6:0] 均为因去歪斜而导致添加的少量时钟周期延迟，单位为时钟周期的 1/66。
stat_rx_rsfc_lane_fill_1	14	输出	RX 通道 1 额外延迟。 每条延迟总线的头 7 个位 [13:7] 即为因去歪斜而导致添加的额外时钟周期延迟数。每条延迟总线的末 7 个位 [6:0] 均为因去歪斜而导致添加的少量时钟周期延迟，单位为时钟周期的 1/66。
stat_rx_rsfc_lane_mapping_*	2	输出	RX 通道映射 位 0：PMA 通道 0 上承载的 FEC 通道的索引 位 1：PMA 通道 1 上承载的 FEC 通道的索引
stat_rx_rsfc_hi_ser_*	1	输出	仅当该核处于旁路指示模式下时，此输出才有效。它表示处于高电平有效时，含 8192 个代码字的窗口内 FEC 符号错误数量超过阈值 K (417)。
stat_rx_rsfc_corrected_cw_inc_*	1	输出	已纠正的代码字数增量
stat_rx_rsfc_uncorrected_cw_inc_*	1	输出	未纠正的代码字数增量
stat_rx_rsfc_error_count0_inc_*	3	输出	通道 0 的符号错误计数增量。
stat_rx_rsfc_error_count1_inc_*	3	输出	通道 1 的符号错误计数增量。

Runtime Switch 信号

在“Configuration”选项卡下选中“Runtime switch”时，本节下所述端口可用。

表 300: Runtime Switch 信号

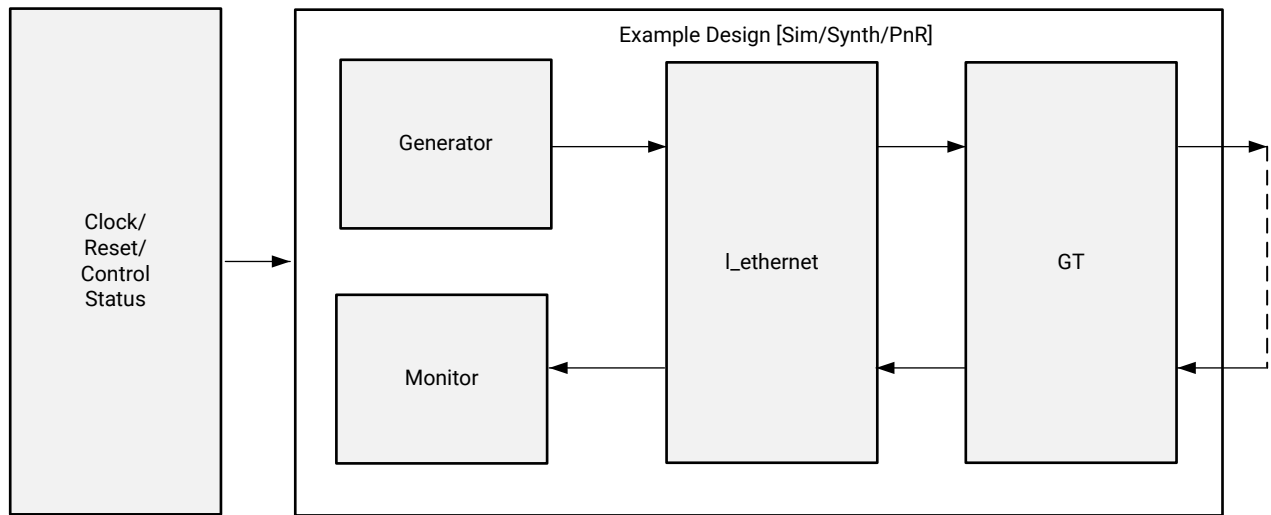
名称	大小	I/O	描述
gt_drp_done_0	1	输入	指示 GT DRP 操作完成。用于复位 GT 模块。
ctl_rate_mode_0	1	输入	此信号导致 IP 核在 50G 运行 (0) 和 40G 运行 (1) 之间切换。请注意，必须针对所选模式纠正时钟频率。
txpllcksel_in_0	8	输入	选择源 TX PLL 时钟以从 GTWIZ IP 生成 TXOUTCLK
rxpllcksel_in_0	8	输入	选择源 RX PLL 时钟以从 GTWIZ IP 生成 RXOUTCLK
txsyscksel_in_0	8	输入	选择用于 GT 通道的源 TX 系统时钟
rxsyscksel_in_0	8	输入	选择用于 GT 通道的源 RX 系统时钟
rxafecfoken_0	4	输入	连接到收发器通道原语上的 RXAFECFOKEN
rxdfecfokfnum_0	16	输入	连接到收发器通道原语上的 RXDFECFOKFCNUM
speed_0	1	输入	此信号用于指示核的处理速度： 1'b1 = 40G 和 1'b0 = 50G
anlt_done_0	1	输出	指示自动协商和链路训练完成 在“Configuration”选项卡下选中“Include AN/LT logic”（包含 AN/LT 逻辑）时，此端口可用。
rxdata_out_0	512	输入	从 GT IP 到 MAC 的 RX 数据总线 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
txdata_in_0	512	输出	从 MAC 到 GT IP 的 TX 数据总线 在“GT Selection and Configuration”选项卡下选中“Include GT subcore in example design”选项时，此端口可用。
axi_ctl_core_mode_switch	1	输出	在“Configuration”选项卡下选中“Include AXI4-Lite”（包含 AXI4-Lite）并将 1 写入 0x013C 自清除寄存器以启动 GT DRP 操作时，此信号可用于在 40G 和 50G 之间切换线速率。
user_reg0	32	输出	来自 AXI4 寄存器映射 user_reg0 寄存器的用户定义的信号。 在“Configuration”选项卡下选中“Include AXI4-Lite”时，此端口可用。

双工操作模式

在此操作模式下，核发射器和接收器处于有效状态，在 GT 输出接口上提供环回，即输出作为输入进行回馈。包生成和监控模块在此模式下同样处于有效状态。生成器模块负责生成所需数量的包，并利用可用的数据接口发射到核。监控模块检查来自接收器的包。

下图显示了双工操作模式。

图 60：双工操作模式



X16598-080621

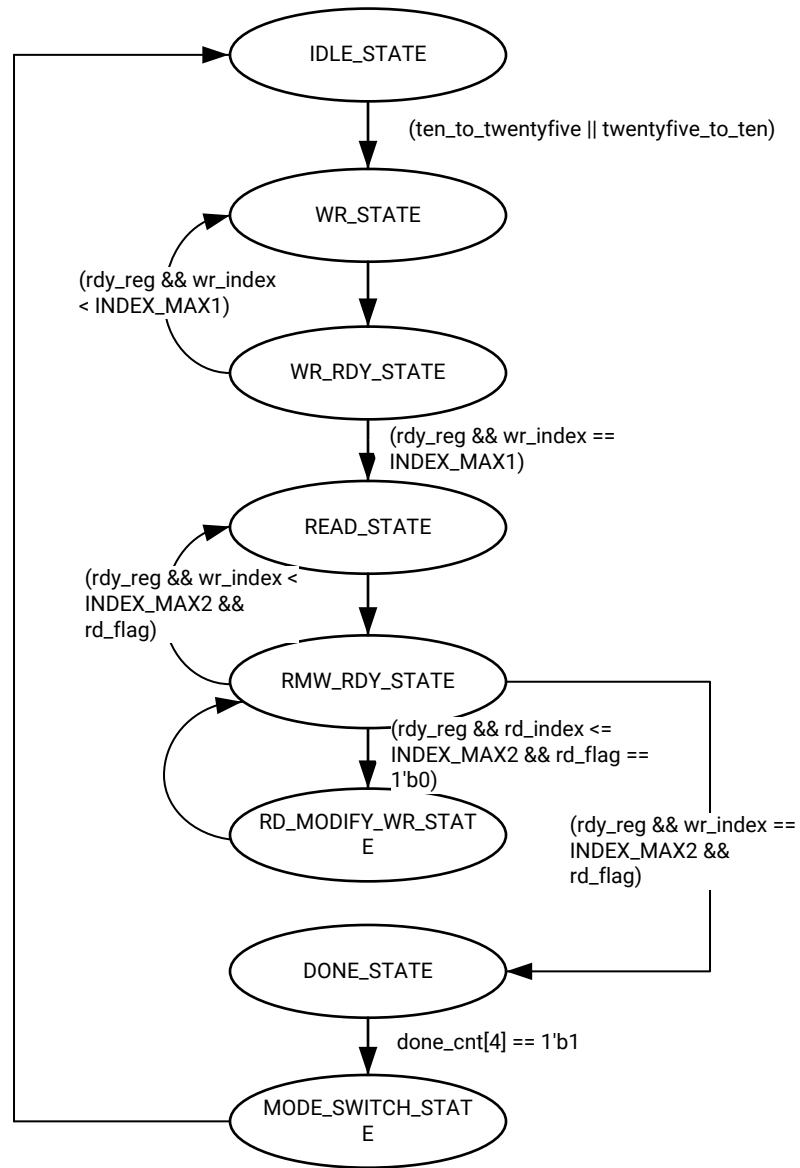
运行时可切换

通过此配置可随时在 40G 和 50G 之间灵活切换线速率。要激活此功能，请在“Configuration”（配置）选项卡下选中“Runtime Switchable mode”（运行时可切换模式）选项复选框。选中此选项时，即可使用 `ch0_txrate` 和 `ch0_rxrate` 端口来更改速度。在内部，GT IP 将执行必要的 DRP 写入操作来达到所需的速度。选中此选项时，设计示例的 `*_pkt_gen_mon.v` 模块内将显示 `*_trans_debug` 模块。此 `*_trans_debug` 模块负责执行所有 GT DRP 写入操作以切换收发器模式，即从 40G 切换至 50G 或从 50G 切换至 40G。将 `mode_change_*` input 信号设置为高电平并保持 2 个时钟周期，然后再将其设置为低电平时，它会为特定的核针对 GT 通道启动 DRP 写入操作，然后将该核复位。仅针对该通道执行 DRP 写入操作。对于线速率 50G，公用端口固定为 QPLL0，线速率 40G 则固定为 QPLL1。

注释：对于切换操作，您应将 `trans_debug` 模块复制到自己的设计中。

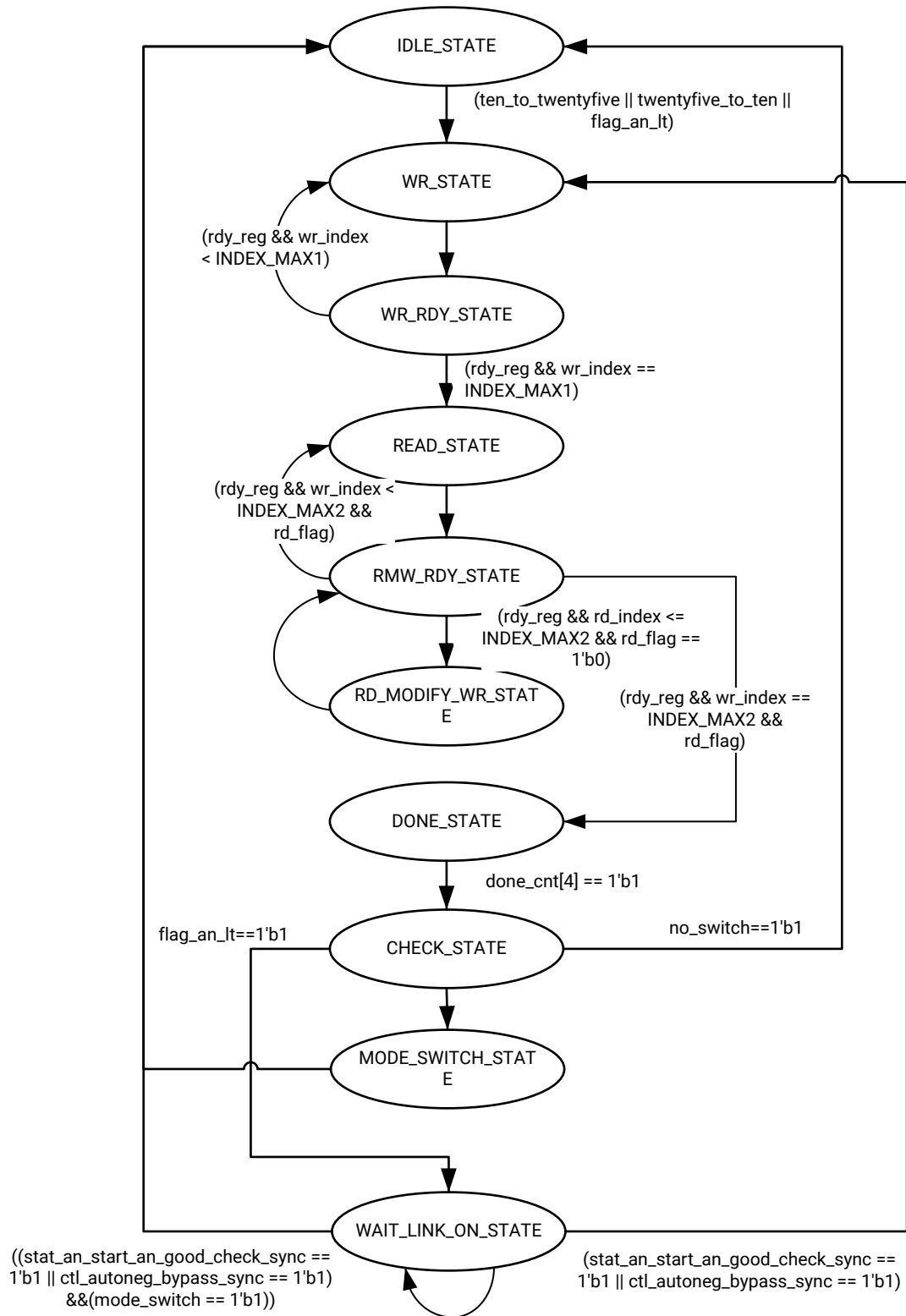
下图显示了在此进程中发生的状态转换。

图 61：不包含 AN/LT 时的运行时可切换 DRP 操作状态转换图



X17802-080621

图 62：包含 AN/LT 时的运行时可切换 DRP 操作状态转换图



X17803-080621

注释：“运行时可切换”小节仅适用于 UltraScale/UltraScale+ 器件。对于 Versal ACAP，您需要通过在 SWITCH_CORE_SPEED_REG: 013C 寄存器中写入 1 来请求速度变更。

共享逻辑实现

共享逻辑包含 GT 公用模块，此模块显示为核的一部分或者显示在设计示例中。

默认情况下，GT 公用模块、复位逻辑和时钟模块显示在 IP 核内部。满足以下条件时，这些模块将布局在核外部，以便与其它设计共享。

- 在“GT Selection and Configuration”（GT 选择和配置）选项卡下选中“Include GT subcore in example design”（在设计示例中包含 GT 子核）选项时。
- 在“Shared Logic”（共享逻辑）选项卡下选中“Include Shared Logic in Example Design”（在设计示例中包含共享逻辑）选项时。

选择设计示例中的共享逻辑时，将在 I_ethernet_*_exdes.v 和 DUT（即，I_ethernet_*.v）之间例化新的 I_ethernet_*_core_support.v 模块。此模块将包含可在多个设计间共享的所有子模块。

下图显示了在单核设计示例内部例化共享逻辑时的实现过程。

图 63：含共享逻辑实现的单核设计示例层级

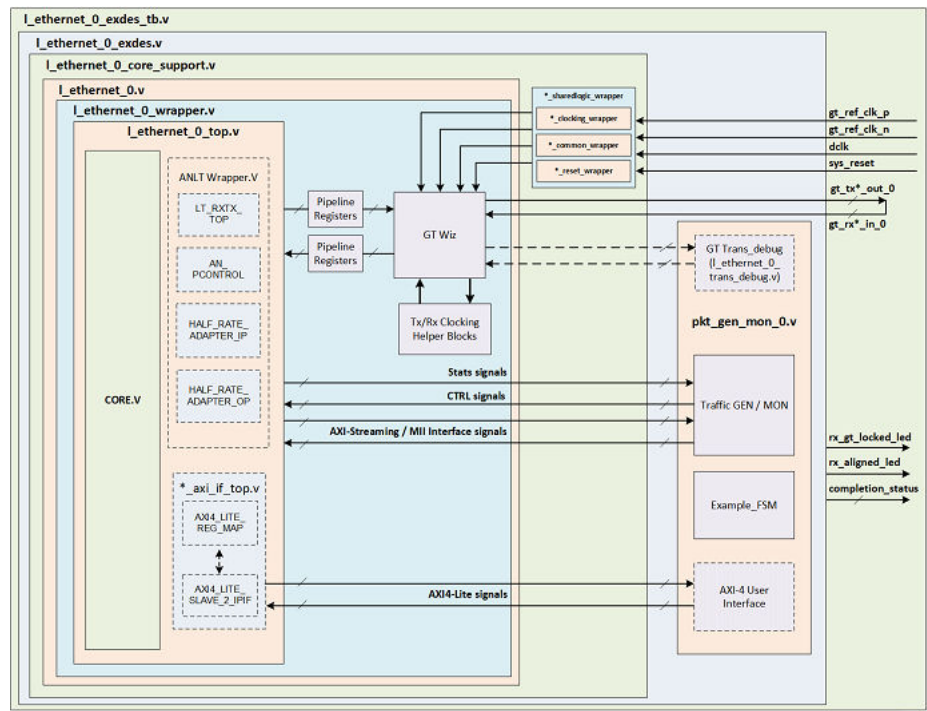
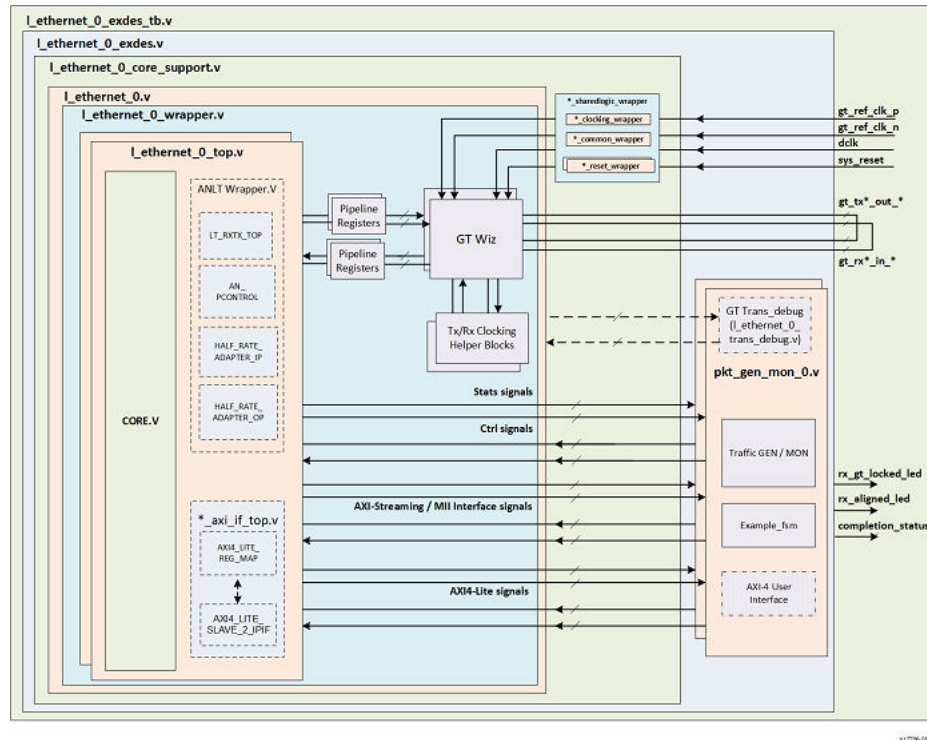


图 64：含共享逻辑实现的多核设计示例层级



下列模块是共享逻辑封装器的一部分。

- *_clocking_wrapper：此模块包含可与其它设计共享的所有时钟资源。
- *_common_wrapper：此模块包含可与其它设计共享的 GT 公用模块。
- *_reset_wrapper：此模块包含适用于用户指定的 Vivado IDE 配置的所有复位逻辑。

AXI4-Lite 接口实现

如果要例化 AXI4-Lite 接口以访问 l_ethernet 核的控制寄存器和状态寄存器，必须启用“Configuration”（配置）选项卡下的“Include AXI4-Lite”（包含 AXI4-Lite）复选框。这样即可启用 l_ethernet_0_axi_if_top 模块（其中包含带有 l_ethernet_0_slave_2_ipif 模块的 l_ethernet_0_pif_registers）。您可从 l_ethernet_0_pkt_gen_mon 模块访问 AXI4-Lite 接口逻辑寄存器（控制寄存器、状态寄存器和统计数据寄存器）。

此模式会启用以下功能：

- 您可通过 AXI4-Lite 接口配置该核的所有控制 (CTL) 端口。此操作的执行方式是将含所需数据的一组地址位置写入寄存器映射接口。
- 您可通过 AXI4-Lite 接口从该核访问所有状态寄存器和统计数据寄存器。此操作的执行方式是通过寄存器映射读取状态寄存器和统计数据寄存器的地址位置。

.h 报头文件

在 Vivado Design Suite 中启用“Include AXI4-Lite”的情况下生成 IP 核时，以头文件 (.h) 格式提供 AXI4 寄存器信息，例如，寄存器地址、寄存器名称（含位的位置）、掩码值、访问类型及其默认值，而且可以在工程路径的 header_files 文件夹下找到头文件。

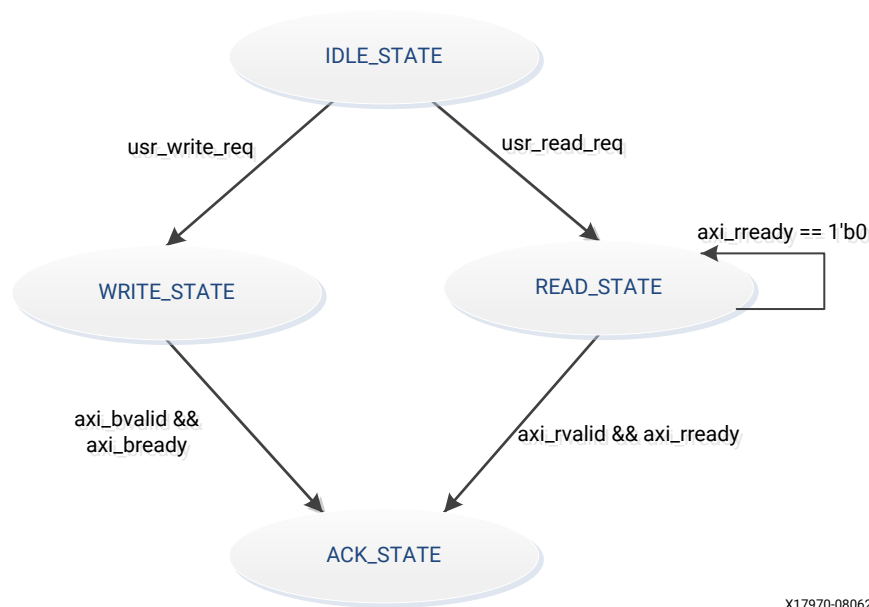
AXI4-Lite 接口用户逻辑

以下章节提供了 AXI4-Lite 接口状态机控制和端口。

用户状态机

通过 AXI4-Lite 从模块接口执行的读写操作由状态机来控制，如下图所示：

图 65：AXI4-Lite 接口的用户状态机



X17970-080621

每个状态的功能描述如下：

- **IDLE_STATE**：默认情况下，FSM 处于 IDLE_STATE 状态。当 `user_read_req` 信号变为高电平 (High) 时，它会转至 READ_STATE 状态，或者如果 `user_write_req` 信号为 High，那么它会转为 WRITE_STATE 状态，否则它将保持 IDLE_STATE 状态。
- **WRITE_STATE**：在此状态下，您提供 `S_AXI_AWVALID`、`S_AXI_AWADDR`、`S_AXI_WVALID`、`S_AXI_WDATA` 和 `S_AXI_WSTRB` 以通过 AXI 写入寄存器映射。当来自 AXI 从接口的 `S_AXI_BVALID` 和 `S_AXI_BREADY` 均为 High 时，它会转至 ACK_STATE。如果在任意违规地址中发生任何写入操作，`S_AXI_BRESP[1:0]` 会指示 2'b10 以断言写入错误信号有效。
- **READ_STATE**：在此状态下，您提供 `S_AXI_ARVALID` 和 `S_AXI_ARADDR` 以通过 AXI 从寄存器映射读取。当 `S_AXI_RVALID` 和 `S_AXI_RREADY` 均为 High 时，它会转至 ACK_STATE。如果在任意违规地址中发生任何读取操作，`S_AXI_RRESP[1:0]` 会指示 2'b10 以断言读取错误信号有效。

- ACK_STATE: 状态将转至 IDLE_STATE。

AXI4-Lite 用户接口端口

表 301: AXI4-Lite 用户接口端口

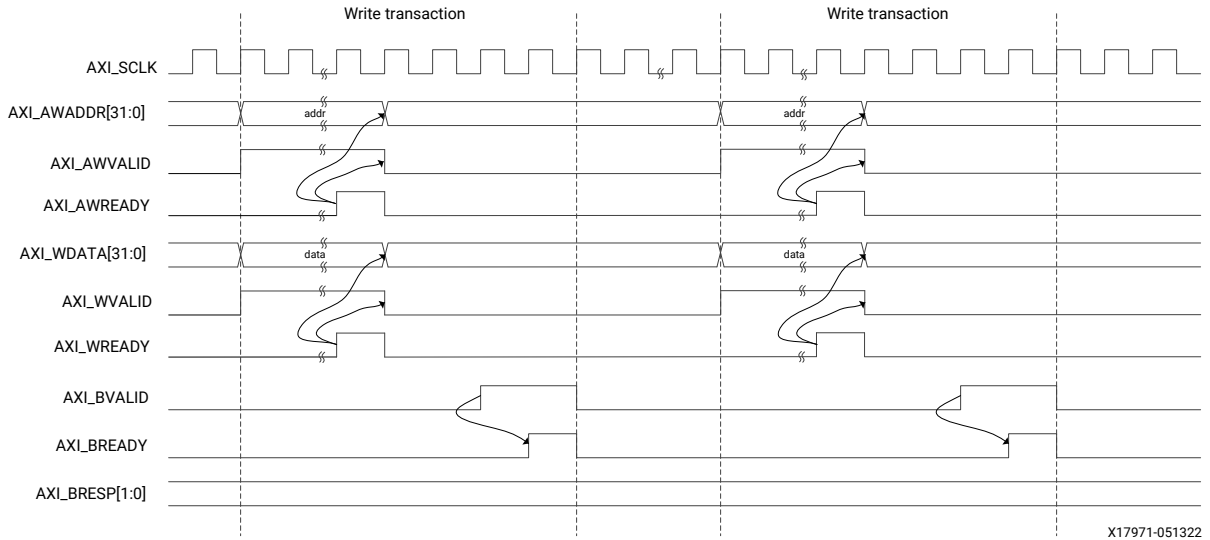
名称	大小	I/O	描述
S_AXI_ACLK	1	输入	AXI 时钟信号
S_AXI_ARESETN	1	输入	AXI 低电平有效同步复位
S_AXI_PM_TICK	1	输入	PM 时钟节拍用户输入
S_AXI_AWADDR	32	输入	AXI 写入地址
S_AXI_AWVALID	1	输入	AXI 写入地址有效
S_AXI_AWREADY	1	输出	AXI 写入地址就绪
S_AXI_WDATA	32	输入	AXI 写入数据
S_AXI_WSTRB	4	输入	AXI 写入选通。此信号用于指示保存有效数据的字节通道。
S_AXI_WVALID	1	输入	AXI 写入数据有效。此信号用于指示有效的写入数据和写入选通均可供使用。
S_AXI_WREADY	1	输出	AXI 写入数据就绪
S_AXI_BRESP	2	输出	AXI 写入响应。此信号用于指示写入传输事务的状态。 'b00 = OKAY 'b01 = EXOKAY 'b10 = SLVERR 'b11 = DECERR
S_AXI_BVALID	1	输出	AXI 写入响应有效。此信号用于指示该通道正在发出有效的写入响应信号。
S_AXI_BREADY	1	输入	AXI 写入响应就绪
S_AXI_ARADDR	32	输入	AXI 读取地址
S_AXI_ARVALID	1	输入	AXI 读取地址有效
S_AXI_ARREADY	1	输出	AXI 读取地址就绪
S_AXI_RDATA	32	输出	AXI 读取数据 (由从接口发出)
S_AXI_RRESP	2	输出	AXI 读取响应。此信号用于指示读取传输的状态。 'b00 = OKAY 'b01 = EXOKAY 'b10 = SLVERR 'b11 = DECERR
S_AXI_RVALID	1	输出	AXI 读取数据有效
S_AXI_RREADY	1	输入	AXI 读取就绪。此信号用于指示用户/主控制器可接受读取数据和响应信息。

用户侧 AXI4-Lite 读写传输事务

下图显示了 AXI4-Lite 接口的时序图波形。

有效的写入传输事务

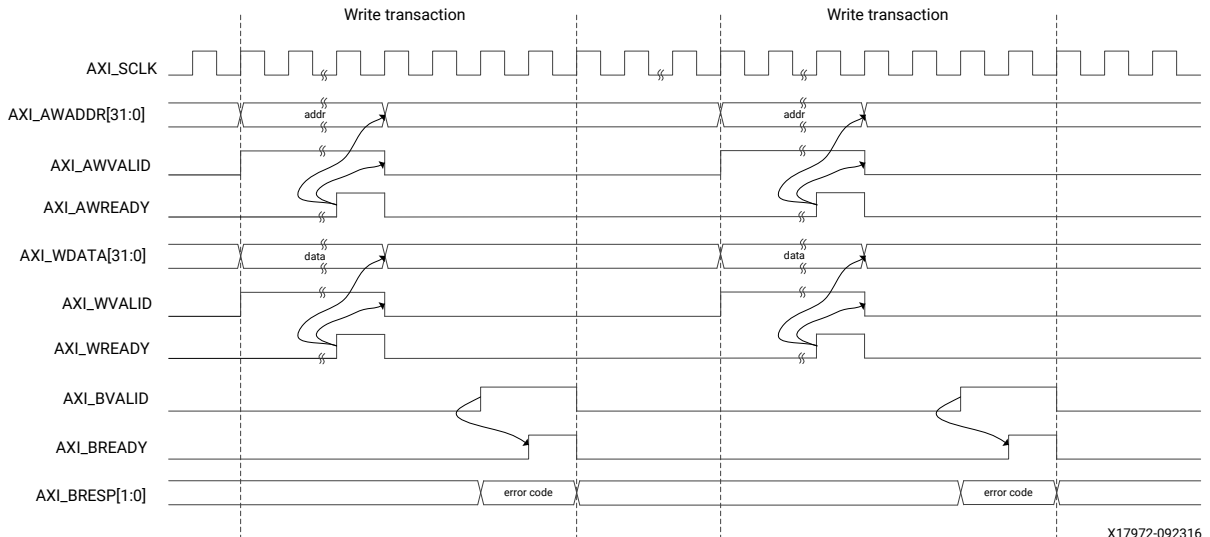
图 66：AXI4-Lite 用户侧写入传输事务



X17971-051322

无效的写入传输事务

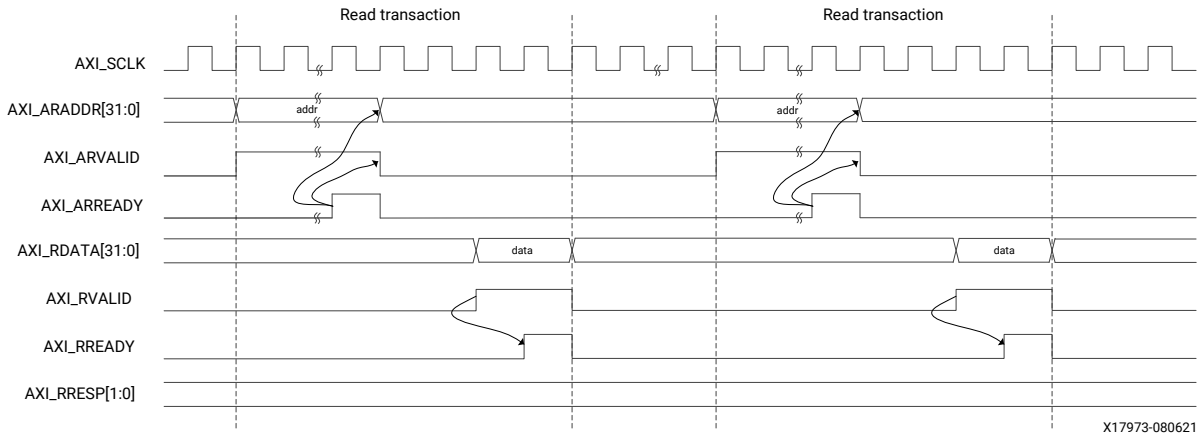
图 67：AXI4-Lite 用户侧写入传输事务（含无效的写入地址）



X17972-092316

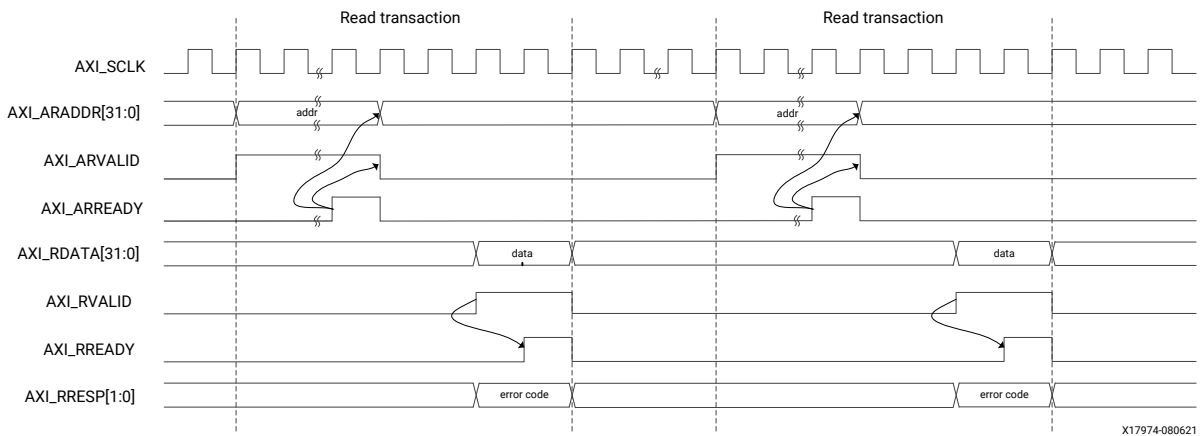
有效的读取传输事务

图 68：AXI4-Lite 用户侧读取传输事务



无效的读取传输事务

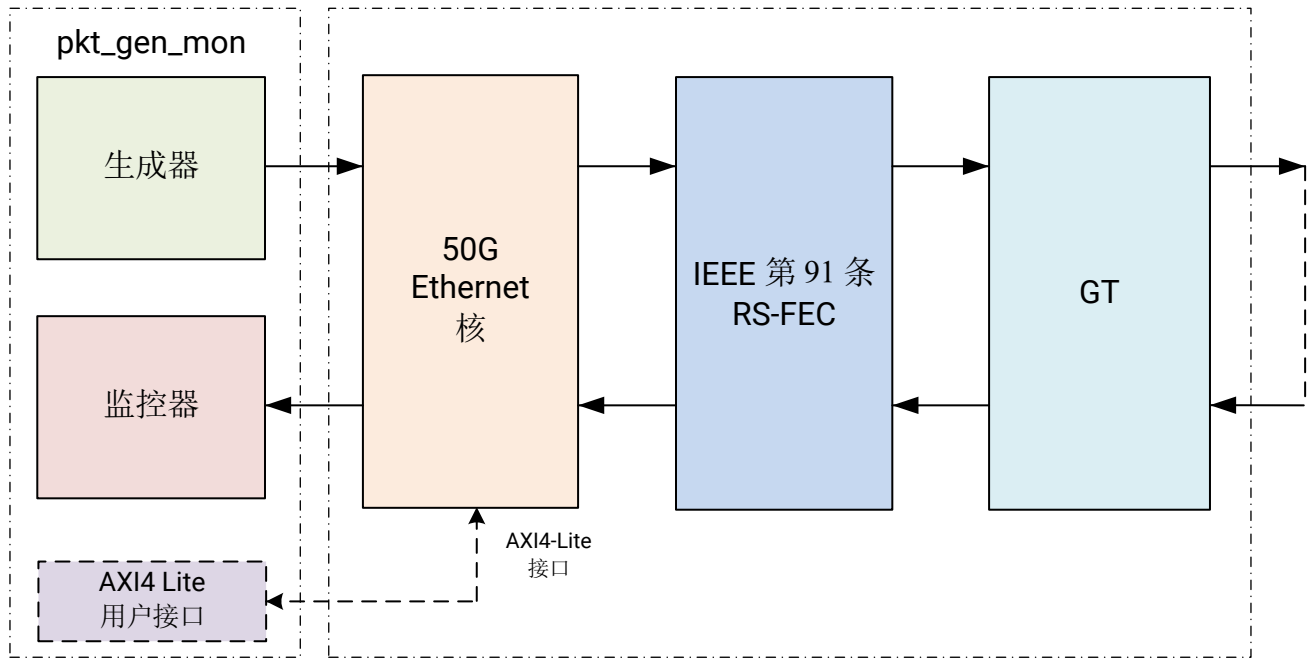
图 69：AXI4-Lite 用户侧读取传输事务（含无效的读取地址）



IEEE 第 91 条 (RS-FEC) 集成

如果想要在 50G Ethernet IP 和 GT 之间包含 IEEE 第 91 条 RS-FEC 软核 IP（用于纠错），则必须在“Configuration”选项卡下选中“Include Clause 91 (RS-FEC)”复选框。此选项仅适用于 50G 速度。

图 70: 50G Ethernet IP 和 GT 之间的 RS-FEC 集成



X17797-042021

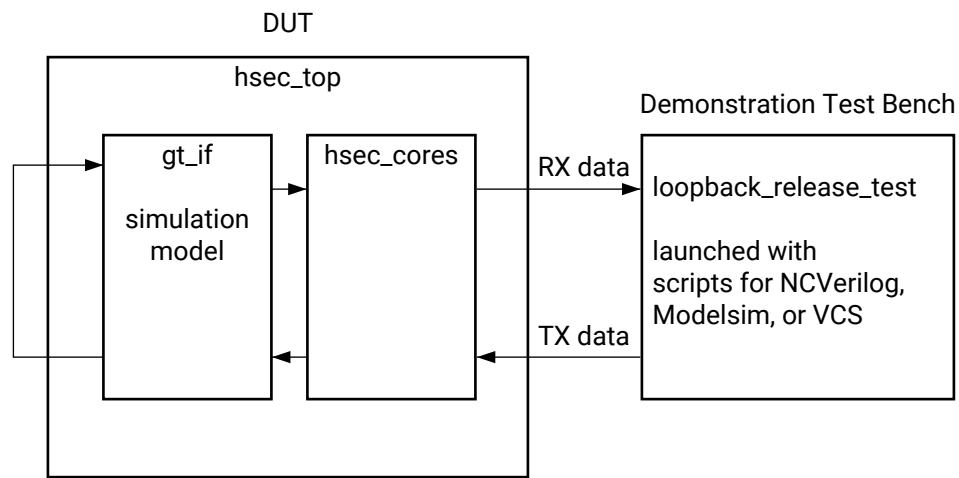
此功能启用 50G 核和 GT 之间例化的 IEEE 第 91 条 RS-FEC 软核 IP 组件。来自 50G 核的 TX SerDes 线将输入到 RS-FEC 软核 IP 以用于前向纠错编码。然后，将来自 RS-FEC 模块的输出馈送到 GT。同样，来自 GT 的 RX SerDes 线将馈送到 RS-FEC 模块以用于纠错解码，然后馈送到 50G 核。

请参阅《50G IEEE 802.3 Reed-Solomon Forward Error Correction LogiCORE IP 产品指南》(PG234) (需注册) 以了解适用于 LogiCORE™ IP 核的 IEEE 第 91 条里德-所罗门前向纠错码 (Reed-Solomon Forward Error Correction) 及其功能。

测试激励文件

每个版本都包含用于演示的测试激励文件，用于在完整的 40G/50G High Speed Ethernet 子系统上执行环回测试。测试程序会测试数据路径以检查能否接收到发射的帧。其中包括 40G/50G High Speed Ethernet 子系统的 RTL 仿真模型。您必须根据所使用的 Vivado® Design Suite 版本中最新的仿真环境设置来为收发器仿真模型提供正确的路径。

图 71: 测试激励文件



X13988-080621

升级

以下章节描述了每个版本间的更改。

从 v2.2 到 v2.3 的更改

端口更改

- 已将端口描述图中的端口名称从 `rx_serdes_resetn` 纠正为 `rx_serdes_reset`。
- 已向 `tx_ptp_tstamp_out[80-1:0]` 和 `tx_ptp_tstamp_out_*` 描述添加了有关双步 1588 操作的文本。
- `ctl_tx_ptp_latency_adjust[10:0]` 信号
删除了有关 802 十进制时钟模式值的语句。
- 更新了 `axi_ctl_core_mode_switch` 以将 `0x018C` 更新为 `0x013C`。
- 已将名称从 `restart_tx_rx_0` 更改为 `restart_tx_rx_*`

新增端口

- `tx_ptp_upd_chksum_in`
- `tx_ptp_pcslane_out`
- `tx_ptp_chksum_offset_in`
- `rx_ptp_pcslane_out`
- `rx_lane_aligner_fill`
- `rx_ptp_tstamp_valid_out`
- `send_continuous_pkts_*`
- `restart_tx_rx_0`
- `mode_change_0`
- `core_speed_0`

移除端口

- `tx_ptp_rxtstamp_in`
- `tx_ptp_rxtstamp_in`

新增寄存器

- STAT_CORE_SPEED_REG: 047C
- user_reg0_*
- RXOUTCLKSEL_IN_*
- TXOUTCLKSEL_IN_*

更改寄存器名称

- 已将 SWITCH_CORE_SPEED_REG: 0180 更新为 SWITCH_CORE_SPEED_REG: 013C。
- 已将 USER_REG_0: 0184 更改为 USER_REG_0: 0138。

删除寄存器

- USER_REG_1: 0188
- CORE_SPEED_REG: 180

从 v2.1 到 v2.2 的更改

新增端口

添加 `gtpowergood_out_*`。

更新端口描述

- `tx_axis_tuser`。
- `dclk`。
- `rx_clk`。
- 分别是: `gt_refclk_p`、`gt_refclk_n` 和 `dclk`。
- `an_clk_*`。

更新寄存器描述

- `ctl_gt_reset_all`。
- `tick_reg`。
- `stat_rx_rsfec_am_lock0` 和 `stat_rx_rsfec_am_lock1`。

从 v2.0 到 v2.1 的更改

新增功能

256 位 AXI4-Stream 数据包接口 (适用于 40Gb/s MAC+PCS) 。

更新功能

自动协商和链路训练的资源利用率已降低。

新增端口

- stat_tx_underflow_err
- stat_tx_overflow_err

更新端口

- tx_axis_tvalid
- tx_mii_reset
- rx_mii_clk
- gt_refclk_out
- tx_mii_d_*
- tx_mii_c_*
- rx_mii_d_*
- rx_mii_c_*
- axi_ctl_core_mode_switch

新增寄存器

STAT_STATUS_REG1: 0408

从 v2.0 (2016 年 10 月 5 日) 到 v2.0 (2016 年 11 月 30 日) 的更改

端口名称更改

- qpll0clk_in 改为 qpll0_clk_in_*
- qpll0refclk_in 改为 qpll0_refclk_in_*
- qpll1clk_in 改为 qpll1_clk_in_*
- qpll1refclk_in 改为 qpll1_refclk_in_*
- gtwiz_reset_qpll0lock_in 改为 gtwiz_reset_qpll0_lock_in_*
- gtwiz_reset_qpll0lock_out 改为 gtwiz_reset_qpll0_lock_out_*
- Stat_rx_rsfc-symbol_error_count_0_inc_* 改为 Stat_rx_rsfc_error_count_0_inc_*
- Stat_rx_rsfc-symbol_error_count_1_inc_* 改为 Stat_rx_rsfc_error_count_1_inc_*
- Stat_tx_rsfc_pcs_block_lock_* 改为 Stat_tx_rsfc_block_lock_*

新增端口

- ctl_tx_ptp_1step_enable_*
- ctl_tx_ptp_latency_adjust_*
- ctl_tx_ptp_vlane_adjust_mode_*
- ctl_ptp_transpclk_mode_*
- tx_ptp_upd_chksum_in_*
- tx_ptp_chksum_offset_in_*
- tx_ptp_rxtstamp_in_*
- stat_an_rxcdrrhold_*
- gt_drp_done_0
- ctl_rate_mode_0
- txpllclkssel_in_0
- rxpllclkssel_in_0
- txsysclkssel_in_0
- rxsysclkssel_in_0
- Rxafecfoken_0
- Rxdfecfokfnum_0
- Speed_0
- anlt_done_0
- rxdata_out_0
- txdata_in_0

从 v1.1 到 v2.0 的更改

新增十六进制地址

- CONFIGURATION_1588_REG: 0038
- CONFIGURATION_1RSFEC_REG: 00D0
- CORE_SPEED_REG: 0180
- USER_REG_0: 0184
- USER_REG_1: 0188
- STAT_RX_RSFEC_STATUS_REG: 0418
- STAT_RX_RSFEC_LANE_FILL_REG1: 0448
- STAT_TX_RSFEC_STATUS_REG: 044C
- STAT_RX_RSFEC_CORRECTED_CW_INC_LSB: 0670

- STAT_RX_RSFECC_CORRECTED_CW_INC_MSB: 0674
- STAT_RX_RSFECC_UNCORRECTED_CW_INC_LSB: 0678
- STAT_RX_RSFECC_UNCORRECTED_CW_INC_MSB: 067C
- STAT_RX_RSFECC_ERR_COUNT0_INC_LSB: 0680
- STAT_RX_RSFECC_ERR_COUNT0_INC_MSB: 0684
- STAT_RX_RSFECC_ERR_COUNT1_INC_LSB: 0688
- STAT_RX_RSFECC_ERR_COUNT1_INC_MSB: 068C

新增寄存器表

- CONFIGURATION_1588_REG: 0038
- CONFIGURATION_1RSFECC_REG: 00D0
- CORE_SPEED_REG: 0180
- USER_REG_0: 0184
- USER_REG_1: 0188
- STAT_RX_RSFECC_STATUS_REG: 0418
- STAT_RX_RSFECC_LANE_FILL_REG1: 0448
- STAT_TX_RSFECC_STATUS_REG: 044C
- STAT_RX_RSFECC_CORRECTED_CW_INC_LSB: 0670
- STAT_RX_RSFECC_CORRECTED_CW_INC_MSB: 0674
- STAT_RX_RSFECC_UNCORRECTED_CW_INC_LSB: 0678
- STAT_RX_RSFECC_UNCORRECTED_CW_INC_MSB: 067C
- STAT_RX_RSFECC_ERR_COUNT0_INC_LSB: 0680
- STAT_RX_RSFECC_ERR_COUNT0_INC_MSB: 0684
- STAT_RX_RSFECC_ERR_COUNT1_INC_LSB: 0688
- STAT_RX_RSFECC_ERR_COUNT1_INC_MSB: 068C

从 v1.0 到 v1.1 的更改

新增端口

新增了以下端口。

- ctl_gt_reset_all_*
- ctl_gt_reset_*
- ctl_gt_rx_reset_*
- gt_reset_all_in_*
- gt_tx_reset_in_*

- gt_rx_reset_in_*
- ctl_an_fec_25g_rs_request
- ctl_an_fec_25g_baser_request
- stat_an_lp_fec_25g_rs_request
- stat_an_lp_fec_25g_baser_request
- tx_ptp_rxtstamp_in
- rx_ptp_pcslane_out_*
- ctl_an_fec_25g_baser_request_*

移除端口

stat_tx_underflow_err

端口更改

- rx_serdes_reset_done_in_* 改为 rx_serdes_reset_*
- 多个 I/O 端口的 SERDES_WIDTH 已更改为 64。
- ctl_an_ability_25gbase_cr 改为 ctl_an_ability_25gbase_krcr
- ctl_an_ability_25gbase_kr 改为 ctl_an_ability_25gbase_krcr_s
- ctl_an_fec_request 改为 ctl_an_fec_10g_request。更新描述。
- stat_rx_errored_block_increment_valid 改为 stat_rx_bad_code_valid
- stat_rx_bad_sh_increment_0[3:0] 改为 stat_rx_framing_err_0[3:0]
- stat_rx_bad_sh_increment_1[3:0] 改为 stat_rx_framing_err_1[3:0]
- stat_rx_bad_sh_increment_2[3:0] 改为 stat_rx_framing_err_2[3:0]
- stat_rx_bad_sh_increment_3[3:0] 改为 stat_rx_framing_err_3[3:0]
- stat_rx_bad_sh_increment_valid_1 改为 stat_rx_valid_1
- stat_rx_bad_sh_increment_valid_2 改为 stat_rx_valid_2
- stat_rx_bad_sh_increment_valid_3 改为 stat_rx_valid_3
- rx_axis_tuser_0 改为 rx_axis_tuser_*

新增十六进制地址

- CONFIGURATION_LT_SEED_REG1: 0114
- CONFIGURATION_LT_COEFFICIENT_REG1: 0134
- STAT_LT_COEFFICIENT1_REG: 0478
- STAT_RX_ERROR_LSB: 0668
- STAT_RX_ERROR_LSB: 066C

寄存器定义中新增信号

- ctl_gt_rx_reset

- `ctl_gt_tx_reset`
- `ctl_tx_ipg_value`
- `ctl_tx_custom_preamble_enable`
- `ctl_rx_custom_preamble_enable`
- `ctl_an_fec_25g_rs_request`
- `ctl_an_fec_25g_baser_request`
- `stat_an_lp_fec_25g_rs_request`
- `stat_an_lp_fec_25g_baser_request`

新增寄存器表

- `CONFIGURATION_LT_SEED_REG1`: 0114
- `CONFIGURATION_LT_COEFFICIENT_REG1`: 0134
- `STAT_LT_COEFFICIENT1_REG`: 0478
- `STAT_RX_ERROR_LSB`: 0668
- `STAT_RX_ERROR_MSB`: 066C

调试

本附录包含有关赛灵思支持网站和调试工具上可用资源的详细信息。

在 Xilinx.com 上寻求帮助

为了帮助您在使用子系统时完成设计和调试进程，[赛灵思技术支持网页](#)上提供了大量关键资源，如产品文档、版本说明、答复记录、已知问题相关信息以及如何获取进一步产品支持的链接。[赛灵思社区论坛](#)还可供会员学习、参与、分享和提出与赛灵思解决方案相关的问题。

文档

本产品指南是与该子系统相关的主要文档。本指南以及有助于设计进程的所有产品相关文档都可以在[赛灵思技术支持网页](#)上找到，也可以通过赛灵思 Documentation Navigator 来获取。要下载赛灵思 Documentation Navigator，请访问[下载页面](#)。如需了解此工具和可用功能的详细信息，请在安装后打开联机帮助。

答复记录

答复记录包括有关常见问题的信息、有关如何解决这些问题的实用信息以及有关赛灵思产品的所有已知问题。我们每天都会创建和维护答复记录，确保用户可以获取最准确的信息。

您可以通过[赛灵思技术支持网页](#)（主页）上的“搜索支持”框找到对应该子系统的答复记录。要最大程度扩展搜索结果范围，请使用关键字，例如：

- 产品名称
- 工具消息
- 所遇到问题的摘要

返回结果后，可以使用过滤器搜索来进一步定位结果。

40G/50G High Speed Ethernet 子系统主答复记录

答复记录 [54690](#)。

技术支持

赛灵思在[赛灵思社区论坛](#)上为此 LogiCORE™ IP 产品提供技术支持，前提是用户按产品文档中所述方式使用该产品。如果您执行以下任何操作，则赛灵思无法保证产品时序和功能的正常运行，也无法保证提供相应支持：

- 在文档中未定义的器件中实现解决方案。

- 超出产品文档中允许的范围自定义解决方案。
- 更改设计中任何标记有“DO NOT MODIFY”的部分。

如需提问，请导航至[赛灵思社区论坛](#)。

调试工具

有许多工具可用于解决 40G/50G High Speed Ethernet 设计问题。至关重要的是要了解哪些工具可用于调试各种情况。

设计示例

High Speed Ethernet IP 核在交付时包含一个附有功能测试激励文件的设计示例网表。该设计包含示例收发器以及适用于通用仿真器程序的环回测试。

Vivado Design Suite 调试功能

Vivado® Design Suite 调试功能可以将逻辑分析器和虚拟 I/O 核直接插入到您的设计中。调试功能还支持您设置触发条件，以便在硬件中捕获应用和集成块端口信号。随后，您便可对捕获的信号进行分析。Vivado IDE 中的这个功能用来对在赛灵思器件中运行的设计进行逻辑调试和确认。

Vivado 逻辑分析器用于与下列逻辑调试 LogiCORE IP 核交互：

- ILA 2.0（及更高版本）
- VIO 2.0（及更高版本）

请参阅《Vivado Design Suite 用户指南：编程和调试》(UG908)。

参考板

各种赛灵思开发板均支持 40G/50G High Speed Ethernet 核。这些开发板可用于进行原型设计并确定核可与系统通信。

建议使用 UltraScale™ 器件来获得最佳性能。确保开发板收发器支持所需的以太网比特率。例如，以下开发板适合许多 UltraScale™ 实现：UltraScale FPGA 评估板 VCU108。

仿真调试

每个 High Speed Ethernet IP 核版本都包含仿真测试激励文件样本。通常其中包括从用户接口的 TX 侧环回、通过 TX 电路、环回到 RX 电路并检查用户接口的 RX 侧接收到的数据包。

每个版本通常包含赛灵思收发器的例化样本，对应于客户所选的器件。环回仿真包括穿过收发器的路径。

仿真是使用专为多个常见的业界标准仿真器所提供的脚本来运行的。

如果仿真未从脚本正常运行，则应检查以下项。

仿真器许可证可用性

如果仿真器无法启动，则表示您可能没有有效许可证。请确保许可证处于最新状态。原因也可能是您的企业拥有可用于某一其它仿真器的许可证，因此，请尝试提供的所有脚本。

库文件位置

每个仿真脚本都会调用所需的赛灵思库文件。这些文件是使用每个版本的 bin 目录中对应的 liblist* 文件来调用的。

在此过程中可能会显示错误消息称仿真器无法找到某些库文件。在此情况下，可能必须修改指向库文件的路径。请与您的 IT 管理员核对，确保路径正确。

版本兼容性

每个版本都根据客户所请求的赛灵思工具版本经过测试。如果仿真未成功完成，您应首先确保将所使用的赛灵思工具版本更新至最新版本。在版本的 README 自述文件中以及该版本所包含的仿真样本 log 日志文件中均指明了首选版本。

仿真缓慢

在某些情况下，仿真可能看上去运行较为缓慢。如果仿真太慢而难以接受，以下建议或有助于提升运行时间性能。

- 使用含更多内存、运行速度更快的计算机。
- 使用平台负载共享工具 (LSF) (如果可在您的组织中使用)。
- 绕过赛灵思收发器 (可能需要您创建自己的测试激励文件)
- 减少发送的数据包的数量。此操作可通过修改提供的测试激励文件样本中的相应参数来实现。
- 缩短对齐标记之间指定的时间。这样应可缩短通道对齐阶段所耗的时间，但是会造成开销增加。但最终在硬件中实现 High Speed Ethernet IP 核时，对齐标记之间的距离应遵循规格要求 (间隔 16,383 个代码字)。

仿真失败且未完成

如果仿真样本在成功完成前即失败并挂起，那么可能发生了超时。请确保针对仿真超时所设置的值足以容纳仿真等待时间段，例如，通道对齐阶段的等待时间段。如需了解更多详情，请参阅 [仿真](#)。

仿真完成但失败

如果仿真样本可完成但失败，请联系赛灵思技术支持。每个版本在交货前都经过测试，通常测试都能成功完成。请参阅仿真样本 log 日志文件，以确认仿真行为是否符合期望的行为。

硬件调试

硬件问题各不相同，可能是链路初始化问题，也可能是测试数小时后才显现的问题。本节提供了常见问题的调试步骤。Vivado® 调试功能是可用于硬件调试的宝贵资源。可以通过调试功能来探测以下各个部分中提到的信号名称，以便对特定问题进行调试。

下列大部分常见问题也可能适用于调试设计仿真。如需了解详细信息，请参阅下列章节：

- [常规检查](#)
- [收发器专用检查](#)
- [以太网专用检查](#)

常规检查

确保核的所有时序约束都已正确整合在设计示例中，并且在实现期间已满足所有约束条件。

- 在布局布线后的时序仿真中是否能够正常工作？如果在硬件中发现问题，但在时序仿真中没有出现问题，则可能表示存在 PCB 问题。确保所有时钟源均处于活动状态且无任何错误。
- 如果在设计中使用 MMCM，请通过监控 `locked` 端口确保所有 MMCM 都被锁定。
- 如果您的输出为 0，请检查您的许可。

收发器专用检查

- 确保 `txn/txp` 和 `rxn/rxp` 线的极性没有反转。如果反转，可使用收发器的 `TXPOLARITY` 和 `RXPOLARITY` 端口进行修复。
- 确认收发器没有保持在复位状态或仍在初始化中。收发器的 `RESETDONE` 输出指示收发器于何时就绪。
- 将收发器置于并行或串行近端环回模式。
- 如果在收发器串行环回模式时操作正常，但在通过光缆执行环回时不正常，则可能表明光学模块存在故障。
- 如果核在收发器并行环回模式下操作正常，但在串行环回模式时不正常，这可能表明收发器有问题。
- 轻微的误码率可以通过调整收发器的发射器预加重和差分摆幅控制属性来解决。

以太网专用检查

在 Ethernet IP 核的首次硬件测试期间，会发生几个常见的问题。这些问题应按以下小节中指示的方式来进行检查。

其中假定 Ethernet IP 核已成功通过硬件中要实现的所有仿真测试。这是执行任何种类的硬件调试的先决条件。

常用调试顺序如下所述：

1. 清理信号完整性。
2. 确保每个 SerDes 均可实现时钟数据恢复 (CDR) 锁定。
3. 检查每个通道是否都已实现字对齐。
4. 检查是否已实现通道对齐。
5. 下一步，继续执行接口和协议调试。

信号完整性

如果首次执行开发板初始化，并且 High Speed Ethernet IP Core 似乎尚未实现通道对齐，那么很可能是信号完整性问题。



重要提示! 必须先解决信号完整性问题，然后才能执行任何其它调试操作。

即使实现了通道对齐，但如果存在周期性的 bip-8 错误，也表明存在信号完整性问题。请检查 bip-8 信号以辅助执行调试。

信号完整性应与 High Speed Ethernet IP 核分开，单独进行调试。应执行以下过程。

注释: 假定 PCB 本身是根据所需的走线阻抗和走线长度（包括 IEEE 以太网标准（IEEE 802.3-2015 标准）中规定的偏差要求）设计和制造的。

- 收发器设置
- 检查噪声
- 误码率测试

如需帮助进行收发器和信号完整性调试，请联系赛灵思技术支持。

通道交换

在以太网中，物理通道可进行交换，协议按正确方式完成通道对齐。因此，通道交换应该不会导致任何问题。

N/P 交换

如果差分信号对的正负信号进行交换，则在此通道上将无法正确接收数据。您应验证每个链路中每个差分对的极性是否正确。

时钟设置和复位

请参阅 [第 4 章：利用子系统进行设计](#) 以查看这些要求。

确保 High Speed Ethernet IP 核与赛灵思收发器参考时钟的时钟频率与订购 IP 核时所请求的配置相匹配。核时钟具有与之关联的最小频率。最大核时钟频率则由时序约束来确定。最小核时钟频率衍生自所需的以太网带宽加上为时钟容限、时钟漂移和时钟抖动所保留的裕度。

调试期间首先需要验证的是确保复位保持断言有效，直至时钟稳定为止。时钟频率必须稳定并且不含毛刺，随后才能使 High Speed Ethernet IP 核解复位。此要求适用于 SerDes 时钟和 IP 核时钟。

如果后续在时钟上检测到任何不稳定现象，则 High Speed Ethernet IP 核必须复位。此类不稳定现象的示例之一是 CDR 锁定丢失。用户逻辑应判定需要复位的所有外部条件（例如，时钟毛刺、CDR 锁定丢失或电源毛刺等）。

当串行数据变为有效后，GT 需要 GTRXRESET 以确保对数据施加正确的 CDR 锁定。在上电、复位或重新连接链路伙伴后，这是必需的。在核级别，为了避免链路 TX 侧发生中断，可使用 `gtwiz_reset_rx_datapath` 触发复位。可通过对光学模块信号丢失使用信号检测或反相（如果可用）来触发复位。如果信号检测或信号丢失不可用，则可添加超时逻辑以监控对齐是否尚未完成，并发出 `gtwiz_reset_rx_datapath` 复位。

除非 IP 核已复位，否则无法执行配置更改。配置更改的示例包括更改最大数据包长度设置。检查端口列表上特定信号的描述，以判定此要求是否适用于要更改的参数。

接口调试

AXI4-Stream 接口

High Speed Ethernet IP 核用户接口被称为 AXI4-Stream。有 2 个版本可用：常规 AXI4-Stream 和跨接式 AXI4-Stream。50 Gb/s 运行仅允许使用跨接式 AXI4-Stream，而 40 Gb/s 运行则提供了 256 位常规 AXI4-Stream 和 128 位跨接式接口的选项。请参阅本指南中的相应章节，了解每个接口的详细描述。

TX 调试（缓冲器错误）

TX 调试是使用多个诊断信号来辅助完成的。

数据必须写入 TX AXI4-Stream 以避免出现上溢或下溢状况。AXI4-Stream 带宽必须始终大于以太网带宽，以避免数据发送发生中断。

将数据写入 AXI4-Stream 时，必须始终对 `tx_rdyout` 信号进行观测。此信号用于指示 TX 缓冲器的填充级别是否在可接受范围内。如果此信号曾断言有效，则必须停止写入 TX AXI4-Stream 直至断言此信号无效为止。

由于 TX AXI4-Stream 带宽大于 TX 以太网接口，因此，此信号被断言有效是很常见的现象，无需担心。当 `tx_rdyout` 断言有效时，必须确保停止 TX 写入。

`tx_rdyout` 断言有效的级别由预先确定的阈值来判定。

在 TX 方向上开始包数据传输事务后，它必须持续运行直至完成，否则可能出现缓冲器下溢（以 `tx_unfout` 信号来标示）。不允许出现此状况；数据必须连续写入 TX AXI4-Stream，不得中断。以太网数据包必须自始至终存在于线上，不得存在间隙或空闲。如果 `tx_unfout` 曾断言有效，则调试必须停止，直至导致下溢的问题得到解决为止。

注释：当此信号采样为 1 时，您必须应用 `tx_reset/sys_reset` 以便使该核从下溢问题恢复。`tx_reset` 仅复位 TX 路径，`sys_reset` 则会恢复整个系统。

RX 调试（缓冲器错误）

如需获取可用于调试 RX 的诊断信号的详细描述，请参阅“端口描述”。

如果根据 IEEE 802.3-2015 标准正确传输了以太网包，则不应出现 RX 错误。但必须首先验证接收到的信号的信号完整性。

`stat_rx_bip_err` 信号用于指示每个通道的信号质量。根据 IEEE 802.3-2015 标准，当误码率过高时，`stat_rx_hi_ber` 信号将断言有效。阈值为 $BER = 10^{-4}$ 。

为帮助调试，可使用 `gt_loopback_in` 信号来执行 GT 近端 PMA 环回。这样会将 TX SerDes 连接到 RX SerDes，从而有效地避免了潜在的信号完整性问题。通过这种方式，接收到的数据可与发射的包进行对比检查，以验证逻辑是否正常运行。

相关信息

[端口描述](#)

协议调试

为了通过 Ethernet IP 核实现无错数据传输，应遵循 IEEE 802.3-2015 标准进行操作。应始终首先确保信号完整性，然后再继续进行协议调试。

对齐标记间隔

根据 IEEE 802.3-2015 标准，针对 TX 和 RX，对齐标记间隔都应设置为 16,383。请检查链路两端是否都编程为该值。

诊断信号

有许多错误指示可用于检查是否存在协议违例。请仔细阅读每项指示的描述，查看它是否适用于特定调试问题。

以下是建议的调试顺序。

1. 请确保已实现字同步。
2. 请确保已实现通道同步（所使用的通道标记对齐字间隔为 16,383 个字）。
3. 验证 bip8 指示符是否已清零。
4. 确保不存在解扰器错误。
5. 消除 CRC32 错误（如果有）。
6. 确保正确遵循 AXI4-Stream 协议进行操作。
7. 确保发送数据包时不存在上溢或下溢状况。

统计数据计数器

实现无错通信时，可监控统计数据指示符以确保流量特性符合期望。部分信号仅为选通，即计数器不包含在 IP 核中。这样您即可对计数器大小进行自定义。计数器为可选。

调试自动协商和链路训练

自动协商

要启用自动协商，请执行以下操作：

- `ctl_autoneg_enable = 1`
- `ctl_autoneg_bypass = 0`

设置 `ctl_an_*` 以播发所期望的自动协商设置。

使用控制和状态接口时，设计示例会将 `ctl_an_*` 值绑定到有效设置。如果使用的是寄存器接口，请参阅“40G/50G High Speed Ethernet 的开发板测试”以了解寄存器顺序。

链路训练

要启用链路训练，请将 `ctl_lt_training_enable` 设置为 1。

- 这样核不会实际执行任何训练。它仅提供第 72.6.10 条所需的控制协议。训练算法由用户负责。
- 该核不会监控 RX 眼，也不会向链路伙伴 TX 发送任何预置、初始化或系数控制请求。建议将 `ctl_lt_rx_trained` 设为 1。设置 `ctl_lt_rx_trained` 即可告知链路伙伴，您的 RX 训练已完成，并且您将不会发送任何其它预置、初始化或系数更改。
- 该核不会对任何 GT TX 振幅或系数控制设置进行调整以响应从链路伙伴接收到的任何训练消息。设计示例链路训练 `Place_Holder` 逻辑可指示何时已达到最大限制。这样应可允许链路训练成功完成。

随机数

`nonce_seed` 必须设置为非零值。

- 如果要连接同一开发板上含相同随机数种子的 2 个端口，那么复位释放时间必须不同。
- 如果 `nonce_seed` 发生更改，则需通过 `an_reset` 来加载新的值。这包括使用 AXI4-Lite 寄存器更改 `nonce_seed`。

下一页信号

如果链路伙伴发送下一页信号，`ctl_an_loc_np_ack` 必须设置为高电平，以确认下一页信号并允许自动协商完成操作。接收到下一页信号或始终绑定到高电平后，此控制信号即可设置为高电平。

有关阶段和状态信号的详细信息

1. 自动协商启动时处于 TX 禁用状态，不显示任何数据，这样可以确保两端链路均处于中断状态。
`stat_an_stat_tx_disable` 信号将进行切换并保持 1 个周期，以指示此阶段开始。
2. TX 禁用状态后，将交换自动协商信息。在此阶段中，`stat_an_rxcdrhold` 保持高电平。
`stat_an_lp_autoneg_able` 和 `stat_an_lp_ability_valid` 信号将切换至高电平并保持 1 个时钟周期，以指示何时 `stat_an_lp*` 信息有效。
3. 链路训练开始时，`stat_an_start_an_good_check` 信号将切换至高电平并保持 1 个时钟周期。
`stat_an_rxcdrhold` 信号将断言无效并切换 `gtwiz_reset_rx_datapath`。链路训练开始后，将采用 500 毫秒定时器来执行训练和块锁定、以任务模式建立连接并完成正常 PCS 操作，否则自动协商将重新启动。当链路训练块完成帧同步后，`stat_lt_frame_lock` 信号会变为高电平，`stat_lt_rx_sof` 将进行切换。在训练帧边界处，`stat_lt_rx_sof` 信号将继续切换至高电平，并保持 1 个时钟周期。
4. 当链路训练完成时，`stat_lt_signal_detect` 信号将断言有效，指示开始正常 PCS 操作。
5. 当块锁定、同步并对齐（对应多通道核），`stat_rx_status` 和 `stat_rx_valid_ctrl_code`（`stat_rx_valid_ctrl_code` 仅适用于单通道 10G/25G 核）转至高电平时，`an_autoneg_complete` 信号会转至高电平。
6. `an_autoneg_complete` 信号必须在 500 ms 超时范围内转至高电平，否则自动协商会重新启动。如果 `stat_rx_status` 在任何时间返回低电平，那么自动协商会重新启动。

仿真与环回

自动协商 TX 禁用状态需耗时 50 毫秒才能完成仿真。使用不含预编译的 IP 库的 `SIM_SPEED_UP` 选项可缩短等待时间。请参阅答复记录 [73518](#) 以获取有关关闭预编译库的更多信息。

自动协商无法在环回过程中完成，因为自动协商要求从链路伙伴接收到的随机数值必须与发送给链路伙伴的随机数值不同。

可添加到 ILA 以供调试的起始信号列表如下所示：

- `sys_reset`
- `an_reset`

- `ctl_an_*`
- `ctl_lt_*`
- `stat_an_start_tx_disable`
- `stat_an_rxcdrhold`
- `stat_an_lp_autoneg_able`
- `stat_an_lp_ability_valid`
- `stat_an_start_an_good_check`
- `stat_lt_frame_lock`
- `stat_lt_signal_detect`
- `stat_lt_link_training`
- `stat_lt_link_training_fail`
- `stat_rx_block_lock`
- `stat_rx_synced` (仅在多通道核上可用)
- `stat_rx_aligned` (仅在多通道核上可用)
- `stat_rx_valid_ctrl_code` (仅在 10G/25G 核上可用)
- `stat_rx_status`
- `stat_rx_bad_code`
- `stat_rx_hi_ber`

如果使用的线速率支持第 74 条法尔码 FEC，则此信号列表如下：

- `stat_fec_inc_cant_correct_count`
- `stat_fec_lock_error`
- `stat_fec_rx_lock`
- `stat_fec_inc_correct_count`
- `ctl_an_fec_10g_request`
- `ctl_fec_rx_enable`
- `ctl_fec_tx_enable`
- `stat_an_fec_enable`
- `stat_an_lp_fec_10g_ability`
- `stat_an_lp_fec_10g_request`

如果使用的线速率支持 RS-FEC，则此信号列表如下：

- `ctl_tx_rsfc_enable`
- `ctl_rx_rsfc_enable`
- `stat_rx_rsfc_am_lock`
- `stat_an_rs_fc_enable`

注释：如果链路伙伴发送下一页信号，那么必须设置 `ctl_an_lo_np_ack` 信号。此端口可绑定到高电平。

相关信息

[使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试](#)

使用 AXI4-Lite 接口调试自动协商和链路训练

要使用 AXI4-Lite 接口调试自动协商和链路训练，请参阅 [使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试](#)。

暂停处理接口

40G/50G High Speed Ethernet 子系统提供了一整套暂停包终止和生成机制。TX 和 RX 具有独立接口用于处理暂停信息，如本附录中所述。

TX 暂停生成

您可以使用 `ctl_tx_pause_req[8:0]` 和 `ctl_tx_pause_enable[8:0]` 输入总线请求发射暂停包。位 [8] 对应于全局暂停包，位 [7:0] 对应于优先暂停包。



重要提示！ 同时请求全局和优先暂停包会导致不可预测的行为，必须避免。

暂停包的内容可使用下列输入管脚来确定。

全局暂停包：

- `ctl_tx_da_gpp[47:0]`
- `ctl_tx_sa_gpp[47:0]`
- `ctl_tx_ethertype_gpp[15:0]`
- `ctl_tx_opcode_gpp[15:0]`
- `ctl_tx_pause_quanta8[15:0]`

优先暂停包：

- `ctl_tx_da_ppp[47:0]`
- `ctl_tx_sa_ppp[47:0]`
- `ctl_tx_ethertype_ppp[15:0]`
- `ctl_tx_opcode_ppp[15:0]`
- `ctl_tx_pause_quanta0[15:0]`
- `ctl_tx_pause_quanta1[15:0]`
- `ctl_tx_pause_quanta2[15:0]`
- `ctl_tx_pause_quanta3[15:0]`
- `ctl_tx_pause_quanta4[15:0]`
- `ctl_tx_pause_quanta5[15:0]`
- `ctl_tx_pause_quanta6[15:0]`

```
· ctl_tx_pause_quanta7[15:0]
```

40G/50G High Speed Ethernet 子系统会自动计算 FCS 并将其添加到包中。对于优先暂停包，40G/50G High Speed Ethernet 子系统还会根据请求的优先级来自动生成启用矢量。

要请求暂停包，必须将 `ctl_tx_pause_req[8:0]` 和 `ctl_tx_pause_enable[8:0]` 总线的对应位设置为 1，并在暂停请求期间使其保持为 1 不变（即，如果这些输入设置为 0，则取消所有挂起的暂停包）。通过发出另一个暂停包并将其中包含的对应暂停量设为 0，即可取消暂停。40G/50G High Speed Ethernet 子系统将在当前进行中的包完成后立即发射暂停包。

为重新发射暂停包，40G/50G High Speed Ethernet 子系统包含总计 9 个独立定时器；对应每个优先暂停包 1 个定时器，对应全局暂停包 1 个定时器。这些定时器将加载对应输入总线的值。发射暂停包后，对应定时器将加载 `ctl_tx_pause_refresh_timer[8:0]` 输入总线的对应值。当定时器超时后，一旦当前进行中的包完成后就会立即发射对应该优先级的另一个包（或全局包）。此外，您可将定时器手动强制设为 0，因此，可针对某一时钟周期将 `ctl_tx_resend_pause` 输入强制设置为 1 来强制重新发射。



重要提示！ 此总线的每个位都必须保持稳定状态达至少 16 个周期，随后才能执行下一次转换。

为了减少暂停包执行优先模式操作的次数，只要任一定时器超时，该暂停包对应的定时器就会被视为“已超时”。此外，等待当前进行中的包完成的过程中，只要任何新定时器发生超时，或者收到您的任何新请求，都将合并为单一暂停帧。例如，如有 2 个定时器正在倒计时，并且您针对第 3 个优先暂停包发送请求，则前 2 个定时器将被强制超时，并且发射当前进行中的包（如有）后就会立即发送一个对应全部 3 个优先级的暂停包。

同样，如果前 2 个定时器中的任何一个定时器超时且未收到您的其它请求，则这 2 个定时器都会被强制超时，并且发射当前进行中的包（如有）后就会立即发送一个对应这 2 个优先级的暂停包。

您可以通过将 `ctl_tx_pause_req[8:0]` 或 `ctl_tx_pause_enable[8:0]` 的相应位设置为 0 来停止生成暂停包。

RX 暂停终止

40G/50G High Speed Ethernet 子系统去除了全局暂停帧和优先暂停帧，并提供了一个简单的握手接口以允许用户响应暂停包。

暂停包判定

暂停包判定分 3 个步骤：

1. 执行检查，确认包属于全局控制包还是优先控制包。

注释： 仅当 `ctl_rx_forward_control` 设为 1 时才会将通过步骤 1 的包转发给您。

2. 如果通过步骤 1，则会检查该包以判定它是否属于全局暂停包。
3. 如果未通过步骤 2，则会检查该包以判定它是否属于优先暂停包。

对于步骤 1，以下伪代码显示了检查功能：

```
assign da_match_gcp = (!ctl_rx_check_mcast_gcp && !ctl_rx_check_ucast_gcp)
|| ((DA
== ctl_rx_pause_da_ucast) && ctl_rx_check_ucast_gcp) || ((DA ==
48'h0180c2000001) &&
ctl_rx_check_mcast_gcp);
```

```

assign sa_match_gcp = !ctl_rx_check_sa_gcp || (SA == ctl_rx_pause_sa);

assign etype_match_gcp = !ctl_rx_check_etype_gcp || (ETYPE ==
ctl_rx_etype_gcp);
assign opcode_match_gcp = !ctl_rx_check_opcode_gcp || ((OPCODE >=
ctl_rx_opcode_min_gcp) && (OPCODE <= ctl_rx_opcode_max_gcp));

assign global_control_packet = da_match_gcp && sa_match_gcp &&
etype_match_gcp &&
opcode_match_gcp && ctl_rx_enable_gcp;

assign da_match_pcp = (!ctl_rx_check_mcast_pcp && !ctl_rx_check_ucast_pcp)
|| ((DA
== ctl_rx_pause_da_ucast) && ctl_rx_check_ucast_pcp) || ((DA ==
ctl_rx_pause_da_mcast) && ctl_rx_check_mcast_pcp);

assign sa_match_pcp = !ctl_rx_check_sa_pcp || (SA == ctl_rx_pause_sa);

assign etype_match_pcp = !ctl_rx_check_etype_pcp || (ETYPE ==
ctl_rx_etype_pcp);

assign opcode_match_pcp = !ctl_rx_check_opcode_pcp || ((OPCODE >=
ctl_rx_opcode_min_pcp) && (OPCODE <= ctl_rx_opcode_max_pcp));

assign priority_control_packet = da_match_pcp && sa_match_pcp &&
etype_match_pcp &&
opcode_match_pcp && ctl_rx_enable_pcp;

assign control_packet = global_control_packet || priority_control_packet;

```

其中 DA 表示目标地址，SA 表示源地址，OPCODE 为操作代码 (opcode)，而 ETYPE 则是从传入的包提取的“ethertype/length”（以太类型/长度）字段。

对于步骤 2，以下伪代码显示了检查功能：

```

assign da_match_gpp = (!ctl_rx_check_mcast_gpp && !ctl_rx_check_ucast_gpp)
|| ((DA
== ctl_rx_pause_da_ucast) && ctl_rx_check_ucast_gpp) || ((DA ==
48'h0180c2000001) &&
ctl_rx_check_mcast_gpp);

assign sa_match_gpp = !ctl_rx_check_sa_gpp || (SA == ctl_rx_pause_sa);
assign etype_match_gpp = !ctl_rx_check_etype_gpp || (ETYPE ==
ctl_rx_etype_gpp);

assign opcode_match_gpp = !ctl_rx_check_opcode_gpp || (OPCODE ==
ctl_rx_opcode_gpp);

assign global_pause_packet = da_match_gpp && sa_match_gpp &&
etype_match_gpp &&
opcode_match_gpp && ctl_rx_enable_gpp;

```

其中 DA 表示目标地址，SA 表示源地址，OPCODE 为操作代码 (opcode)，而 ETYPE 则是从传入的包提取的“ethertype/length”（以太类型/长度）字段。

对于步骤 3, 以下伪代码显示了检查功能:

```
assign da_match_ppp = (!ctl_rx_check_mcast_ppp && !ctl_rx_check_ucast_ppp)
&& ((DA
== ctl_rx_pause_da_ucast) && ctl_rx_check_ucast_ppp) || ((DA ==
ctl_rx_pause_da_mcast) && ctl_rx_check_mcast_ppp);

assign sa_match_ppp = !ctl_rx_check_sa_ppp || (SA == ctl_rx_pause_sa);
assign etype_match_ppp = !ctl_rx_check_etype_ppp || (ETYPE ==
ctl_rx_etype_ppp);

assign opcode_match_ppp = !ctl_rx_check_opcode_ppp || (OPCODE ==
ctl_rx_opcode_ppp);
assign priority_pause_packet = da_match_ppp && sa_match_ppp &&
etype_match_ppp &&
opcode_match_ppp && ctl_rx_enable_ppp;
```

其中 DA 表示目标地址, SA 表示源地址, OPCODE 为操作代码 (opcode), 而 ETYPE 则是从传入的包提取的“ethertype/length” (以太类型/长度) 字段。

用户接口

将使用简单的握手协议来告知您, 已使用 `ctl_rx_pause_enable[8:0]`、`stat_rx_pause_req[8:0]` 和 `ctl_rx_pause_ack[8:0]` 总线接收到暂停包。对于这些总线, 位 [8] 对应于全局暂停包, 位 [7:0] 对应于优先暂停包。

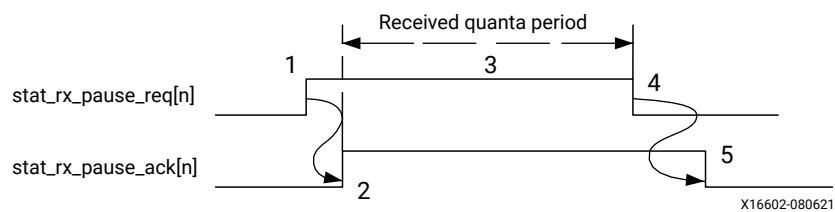
收到暂停包时, 会发生以下步骤:

1. 如果 `ctl_rx_pause_enable[8:0]` 的对应位为 0, 那么将忽略暂停量并且 40G/50G High Speed Ethernet 子系统保持处于步骤 1 不变。否则, `stat_rx_pause_req[8:0]` 总线的对应位将设置为 1, 并且接收到的暂停量将加载到定时器中。
注释: 如果当暂停处理处于步骤 2 或后续阶段时, `ctl_rx_pause_enable[8:0]` 的任一位设置为 0 (即禁用), 那么 40G/50G High Speed Ethernet Subsystem 会照常完成这些步骤, 然后返回步骤 1。
2. 如果 `ctl_rx_check_ack` 输入为 1, 那么 40G/50G High Speed Ethernet 子系统会等待您将 `ctl_rx_pause_ack[8:0]` 总线的相应位设置为 1。
3. 当您 `ctl_rx_pause_ack[8:0]` 的适当位设置为 1 之后或者如果 `ctl_rx_check_ack` 为 0, 那么 40G/50G High Speed Ethernet 子系统会开始对定时器进行倒计时。
4. 当定时器超时, 40G/50G High Speed Ethernet 子系统会将 `stat_rx_pause_req[8:0]` 的相应位重新设置为 0。
5. 如果 `ctl_rx_check_ack` 输入为 1, 则当您将 `ctl_rx_pause_ack[8:0]` 的相应位重新设为 0 时, 此操作即告完成。

如果您不将 `ctl_rx_pause_ack[8:0]` 的相应位重新设为 0, 那么 40G/50G High Speed Ethernet 子系统会在 32 个时钟周期后将此操作视为完成。

下图中演示了前述步骤, 每个步骤均显示在波形上。

图 72: RX 暂停接口示例



如果步骤 2 到 5 期间任意时间收到新的暂停包，那么定时器将加载新获取的暂停量值，并且此进程继续运行。

附加资源与法律声明

赛灵思资源

如需获取答复记录、技术文档、下载以及论坛等支持性资源，请参阅[赛灵思技术支持](#)。

Documentation Navigator 与设计中心

赛灵思 Documentation Navigator (DocNav) 提供了访问赛灵思文档、视频和支持资源的渠道，您可以在其中筛选搜索信息。要打开 DocNav，请执行以下操作：

- 在 Vivado® IDE 中，单击“Help” → “Documentation and Tutorials”。
- 在 Windows 中，单击“Start” → “All Programs” → “Xilinx Design Tools” → “DocNav”。
- 在 Linux 命令提示中输入 `docnav`。

赛灵思设计中心提供了根据设计任务和其它主题整理的文档链接，可供您用于了解关键概念以及常见问题解答。要访问设计中心，请执行以下操作：

- 在 DocNav 中，单击“Design Hubs View”选项卡。
- 在赛灵思网站上，查看[设计中心](#)页面。

注释：如需了解有关 DocNav 的更多信息，请参阅赛灵思网站上的 [Documentation Navigator](#) 页面。

参考资料

以下技术文档是非常实用的补充资料，可配合本指南一起使用：

1. IEEE 以太网标准 ([IEEE 802.3-2015 标准](#))
2. 25G 和 50G 以太网联盟 Schedule 3 v1.6 (2015 年 8 月 18 日) (<http://25gethernet.org/>)
3. 《UltraScale 架构 GTH 收发器用户指南》([UG576](#))
4. 《UltraScale 架构 GTY 收发器用户指南》([UG578](#))
5. 《Vivado Design Suite 用户指南：采用 IP integrator 设计 IP 子系统》([UG994](#))

6. 《Vivado Design Suite 教程：采用 IP integrator 设计 IP 子系统》(UG995)
7. 《Vivado Design Suite 用户指南：采用 IP 进行设计》(UG896)
8. 《Vivado Design Suite 用户指南：入门指南》(UG910)
9. 《Vivado Design Suite 用户指南：逻辑仿真》(UG900)
10. 《ISE 到 Vivado Design Suite 移植指南》(UG911)
11. 《Vivado Design Suite 用户指南：编程和调试》(UG908)
12. 《50G IEEE 802.3 Reed-Solomon Forward Error Correction LogiCORE IP 产品指南》(PG234) (需注册方可访问)
13. 《AMBA AXI4-Stream 协议规范 V1.0》(Arm IHI 0051A)
14. IEEE 1588-2008 标准《IEEE 网络测量和控制系统的精密时钟同步协议标准》(standards.ieee.org/findstds/standard/1588-2008.html)

修订历史

下表列出了本文档的修订历史。

章节	修订综述
2022 年 11 月 3 日 3.3 版	
AXI4-Stream 接收接口	更新本章节中的时钟域名称
正常帧接收	更新章节
AXI4-Stream 发射接口	更新章节
2022 年 5 月 16 日 3.3 版	
配置寄存器	将 reg 0154 重命名为 GT_WIZ_CONTROL_REG。
GT_WIZ_CONTROL_REG : 0154	添加位 3 到位 22。
公共时钟/复位信号	新增端口。
2021 年 10 月 27 日 3.2 版	
不适用	更新 Versal GTM 支持
适用于 Versal ACAP 的常用收发器端口	更新本章节中的表格。
MODE_REG: 0008	更新表格。
STAT_GT_WIZ_REG: 04A0	更新表格。
适用于 GTM 的 SerDes 数据映射	更新表格。
2021 年 8 月 9 日 3.2 版	
整个文档	编辑更新。
2021 年 8 月 6 日 3.2 版	
RX 和 TX PCS 通道标记值	更新本章节中的表格。
“GT Selection and Configuration” 选项卡	更新图像和表格。
RX 路径控制信号、状态信号和统计数据信号	更新本章节中的表格。
TX 暂停接口控制信号、状态信号和统计数据信号	更新本章节中的表格。
RX 暂停接口控制信号、状态信号和统计数据信号	更新本章节中的表格。
IEEE 1588 TX/RX 接口控制信号、状态信号和统计数据信号	更新本章节中的表格。

章节	修订综述
第 91 条 RS-FEC 接口控制信号、状态信号和统计数据信号	更新主题
2021 年 2 月 4 日 3.2 版	
常规更新	新增对 Versal ACAP 的支持
2020 年 6 月 3 日 3.1 版	
使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试	更新主题
LogiCORE 设计示例时钟设置和复位	更新图
自定义和生成子系统	更新 Vivado 图形
设计示例层级 (设计示例中包含 GT)	更新图形
调试自动协商和链路训练 使用 AXI4-Lite 接口调试自动协商和链路训练	新增调试主题
2019 年 10 月 30 日 3.0 版	
使用 AXI4-Lite 接口对 40G/50G High Speed Ethernet 执行开发板测试	新增章节
2019 年 5 月 22 日 2.5 版	
整个文档	更新 GTM 时钟和复位图。
第 4 章: 利用子系统进行设计	新增表格
2018 年 12 月 5 日 2.4 版	
整个文档	新增 GTM 时钟和复位图。 RSFEC 条款已从 108 更新为 91。
2018 年 4 月 4 日 2.3 版	
时延	更新表格 新增有关 <code>ctl_autoneg_bypass</code> 信号的注释以及有关 <code>runtime_switchable</code> 信号的描述。
第 5 章: 设计流程步骤	更新图示 为 GT RefClk (以 MHz 为单位) 选项新增表注释 修订错别字: “udp” 已替换为 “upd”
AXI4-Lite 接口实现	新增 .h 头文件描述
2017 年 12 月 20 日 2.3 版	
请阅读: 重要法律声明	更新
第 3 章: 产品规格	已将 <code>ctl_tx_test_pattern</code> 描述中的 3.42.3 更新为 3.42.7 新增有关表中无效前导码的注释 更新 <code>stat_rx_bad_preamble</code> 的描述
第 6 章: 设计示例	新增有关表中无效前导码的注释 更新 <code>send_continuous_pkts_*</code> 的描述
2017 年 10 月 4 日 2.3 版	
第 2 章: 概述	更新表格
第 3 章: 产品规格	新增 RX 和 TX 时延值 更新图示 在“连续传输”部分的第一段中新增澄清文本 在“统计数据计数器”部分中新增有关 <code>STAT_*_MSB/LSB</code> 寄存器的注释 在 <code>SWITCH_CORE_SPEED_REG: 018C</code> 表下方新增有关“Runtime Switch”模式的文本。
出口	新增时延值

章节	修订综述
第 5 章: 设计流程步骤	更新屏幕显示 从表中移除“AN/LT 时钟”选项。
从 v2.2 到 v2.3 的更改	新增章节
2017 年 6 月 7 日 2.2 版	
许可和订购	更新订购信息
时延	更新表。已将多个信号的 64 字节替换为 16 字节。
第 4 章: 利用子系统进行设计	更新屏幕显示
从 v2.1 到 v2.2 的更改	新增章节
2017 年 4 月 5 日 2.1 版	
功能特性	新增两项新功能
IP 相关信息	更新“支持的用户接口”行, 在其中新增 128 位跨接接口
第 2 章: 概述	更新
标准	更新
第 3 章: 产品规格	新增图示
第 5 章: 设计流程步骤	更新图示
TX 暂停生成	新增重要注释
参考资料	更新
2016 年 11 月 30 日 2.0 版	
第 3 章: 产品规格	已将“AXI4-Lite 寄存器空间”的“配置寄存器”小节第二句中的“tx_reset 和 rx_reset”更改为“s_axi_aresetn”并将“高电平有效”更改为“低电平有效”
第 5 章: 设计流程步骤	更新表格, 添加表注释
第 6 章: 设计示例	在“状态寄存器”部分中新增有关清除状态寄存器的文本 在“统计数据计数器”部分中新增有关清除统计数据计数器的文本
2016 年 10 月 5 日 2.0 版	
整个文档	新增 tick_reg_mode_sel 参考
第 4 章: 利用子系统进行设计	更新图示
第 5 章: 设计流程步骤	更新图示
RS-FEC 支持	新增章节
设计示例层级 (设计示例中包含 GT)	新增章节
时延	新增章节
运行时可切换	新增章节
第 3 章: 产品规格	新增寄存器信息
2016 年 6 月 8 日 1.1 版	
第 3 章: 产品规格	更新图示
第 4 章: 利用子系统进行设计	更新图示
第 5 章: 设计流程步骤	更新图示
附录 A: 升级	新增端口并修改端口名。
整个文档	已将“HSEC”更改为“40G/50G High Speed Ethernet Subsystem” 已通篇将 50GMII 更改为 XL/50GMII
TX 调试 (缓冲器错误)	更新了描述

章节	修订综述
2016 年 4 月 6 日 1.0 版	
初始版本	不适用

请阅读：重要法律声明

本文向贵司/您所提供的信息（下称“资料”）仅在对赛灵思产品进行选择和使用参考。在适用法律允许的最大范围内：(1) 资料均按“现状”提供，且不存在任何瑕疵，赛灵思在此声明对资料及其状况不作任何保证或担保，无论是明示、暗示还是法定的保证，包括但不限于对适销性、非侵权性或任何特定用途的适用性的保证；且 (2) 赛灵思对任何因资料发生的或与资料有关的（含对资料的使用）任何损失或赔偿（包括任何直接、间接、特殊、附带或连带损失或赔偿，如数据、利润、商誉的损失或任何因第三方行为造成的任何类型的损失或赔偿），均不承担责任，不论该等损失或者赔偿是何种类或性质，也不论是基于合同、侵权、过失或是其它责任认定原理，即便该损失或赔偿可以合理预见或赛灵思事前被告知有发生该损失或赔偿的可能。赛灵思无义务纠正资料中包含的任何错误，也无义务对资料或产品说明书发生的更新进行通知。未经赛灵思公司的事先书面许可，贵司/您不得复制、修改、分发或公开展示本资料。部分产品受赛灵思有限保证条款的约束，请参阅赛灵思销售条款：<https://china.xilinx.com/legal.htm#tos>；IP 核可能受赛灵思向贵司/您签发的许可证中所包含的保证与支持条款的约束。赛灵思产品并非为故障安全保护目的而设计，也不具备此故障安全保护功能，不能用于任何需要专门故障安全保护性能用途。如果把赛灵思产品应用于此类特殊用途，贵司/您将自行承担风险和法律责任。请参阅赛灵思销售条款：<https://china.xilinx.com/legal.htm#tos>。

关于与汽车相关用途的免责声明

如将汽车产品（部件编号中含“XA”字样）用于部署安全气囊或用于影响车辆控制的应用（“安全应用”），除非有符合 ISO 26262 汽车安全标准的安全概念或冗余特性（“安全设计”），否则不在质保范围内。客户应在使用或分销任何包含产品的系统之前为了安全的目的全面地测试此类系统。在未采用安全设计的条件下将产品用于安全应用的所有风险，由客户自行承担，并且仅在适用的法律法规对产品责任另有规定的情况下，适用该等法律法规的规定。

版权声明

© 2014-2022 年 Advanced Micro Devices, Inc. 版权所有。Xilinx、赛灵思徽标、Alveo、Artix、Kintex、Kria、Spartan、Versal、Vitis、Virtex、Vivado、Zynq 及本文提到的其它指定品牌均为赛灵思在美国及其它国家或地区的商标。“AMBA”、“AMBA Designer”、“Arm”、“ARM1176JZ-S”、“CoreSight”、“Cortex”、“PrimeCell”、“Mali”和“MPCore”为 Arm Limited 在欧盟及其它国家或地区的注册商标。“PCI”、“PCIe”和“PCI Express”均为 PCI-SIG 拥有的商标，且经授权使用。所有其它商标均为各自所有方所属财产。