

Logos2 系列 FPGA 时钟资源(Clock)

用户指南

(UG040004, V1.2)

(2023.07.31)

深圳市紫光同创电子有限公司

版权所有 侵权必究

文档版本修订记录

版本号	发布日期	修订记录
V0.1	2019.12.30	初始版本
V0.2	2020.3.30	<ol style="list-style-type: none"> 修正 GPLL 和 PPLL 中 APB 端口列表说明, 针对 APB 复位信号和 APB 使能信号的描述发生变动 “GPLL 使用说明相位调整”小节修正对 GPLL 相位调整的使用描述, GPLL 相位调整支持静态调整和动态配置 (通过 APB 接口配置或 DPS 接口配置) “PPLL 使用说明相位调整”小节修正 PPLL 相位调整描述, PPLL 相位调整支持静态调整和动态配置 (通过 APB 接口)
V1.0	2020.11.20	<ol style="list-style-type: none"> 新增 GPLL “扩频时钟输入和产生”小节图横坐标标识, 英文改成中文
V1.1	2021.10.13	<ol style="list-style-type: none"> 新增 GPLL APB 配置接口时序图图 2-32、图 2-33 和图 2-35, 新增 PPLL APB 配置接口时序图图 2-53、图 2-54 和图 2-56 更新 GPLL “相位调整”小节和 PPLL “相位调整”小节描述 修正 GPLL “环路带宽”小节和 PPLL “环路带宽”小节描述, 删除第 (3) 点 总体介绍: 时钟资源分 2 小节说明, 明确时钟管脚到不同 buffer 的限制, 新增时钟连接概述小节 详细介绍: 新增 1.2 小节输入时钟管脚布局规则表, 增加全局时钟 GTP、水平时钟 GTP、区域时钟 GTP、IO 时钟 GTP、跨区域时钟 GTP 小节, 更新“全局时钟网络”、“水平时钟网络”、“区域时钟网络”、“IO 时钟网络”、“跨区域时钟网络”和“PLL 时钟连接”节小示意图以及相关描述 新增 GTP GPLL 配置约束小节 新增 GPLL 和 PPLL 输入时钟丢失复位要求描述, 新增上电无输入时钟时输出时钟状态描述 新增 IO 时钟 GTP 参数描述表 2-16 修正图 2-39 和图 2-60, 新增相关描述 新增 GPLL 和 PPLL 输出频率可编程原理描述 更新所有性能参数指标引用文档描述
V1.2	2023.07.31	<ol style="list-style-type: none"> 更新图 2-2 时钟管脚连接示意图, 删除一对 GMCLK 和 GSCLK 悬空管脚, 更新相关文字描述; 修正图 2-4 和图 2-5, 并新增相关描述 将端口描述表 2-2 中 SIM_DEVICE 参数更新到参数描述表 2-3 总体介绍新增概述, 删除所有 GTP 参数列表描述的默认值, 新增 GTP 信息来源描述 更新 GPLL 关电时序相关描述; 修正图 2-37 GTP_GPLL 复位时序和图 2-38 GTP_GPLL 关电时序; 新增 RST 和 PLL_PWD 释放后到 GPLL LOCK 变成高电平之前输出时钟状态描述; 删除 TPLL_PWD 指标 更新 PPLL 关电时序相关描述; 修正图 2-58 GTP_PPLL 复位时序和图 2-59 GTP_PPLL 关电时序; 新增 RST 和 PLL_PWD 释放后到 PPLL LOCK 变成高电平之前输出时钟状态描述; 删除 TPLL_PWD 指标 新增 GPLL/PPLL 输出时钟 CLKOUT0N~3N 可单独驱动时钟 buffer 备注 新增 GTP_CLKBUFGMUX_E1 异常配置注意事项 删除 GTP_CLKBUFX、GTP_CLKBUFR、GTP_IOCLKBUF 和 GTP_CLKBUFM 时序图

术语与缩略语

Terms and Abbreviations 术语与缩略语	Full Spelling 英文全拼	Chinese Explanation 中文解释
USCM	User Select Clock Mux	全局时钟缓冲器
SRB	Signal Relay Block	信号互联模块
HCKB	Horizontal Clock Buffer	水平时钟缓冲器
RCKB	Regional Clock Buffer	区域时钟缓冲器
MRCKB	Multi-Region Clock Buffer	跨时钟区域的时钟缓冲器
GMCLK	Global Multiplex Clock	多区域全局输入时钟
GSCLK	Global Single Clock	单区域全局输入时钟
GTP	General Technology Primitive	通用原语

目 录

文档版本修订记录	1
术语与缩略语	2
目 录	3
表目录	5
图目录	7
1 总体介绍	10
1.1 时钟资源概述	10
1.2 时钟连接概述	12
1.3 CLOCK REGION	14
2 详细介绍	15
2.1 时钟管脚资源	15
2.1.1 时钟管脚介绍	15
2.1.2 时钟输入管脚布局规则	16
2.2 全局时钟资源	18
2.2.1 全局时钟网络	18
2.2.2 全局时钟 GTP	19
2.3 水平时钟资源	31
2.3.1 水平时钟网络	31
2.3.2 水平时钟 GTP	32
2.4 区域时钟资源	34
2.4.1 区域时钟网络	34
2.4.2 区域时钟 GTP	35
2.5 IO 时钟的资源	37
2.5.1 IO 时钟网络	37
2.5.2 IO 时钟 GTP	38
2.6 跨区域时钟资源	39
2.6.1 跨区域时钟网络	39

2.6.2 跨区域时钟 GTP	40
2.7 PLL 时钟连接	43
2.8 GPLL 使用说明	44
2.8.1 GPLL 概述	44
2.8.2 GTP_GPLL 顶层框图	46
2.8.3 GTP_GPLL 端口列表	46
2.8.4 GTP_GPLL 参数列表	48
2.8.5 GTP_GPLL APB 接口配置	50
2.8.6 GTP_GPLL 功能说明	55
2.8.7 GTP_GPLL 配置约束	65
2.8.8 GTP_GPLL 应用模式	66
2.9 PPLL 使用说明	70
2.9.1 PPLL 概述	70
2.9.2 GTP_PPLL 顶层框图	72
2.9.3 GTP_PPLL 端口列表	72
2.9.4 GTP_PPLL 参数列表	73
2.9.5 GTP_PPLL APB 接口配置	75
2.9.6 GTP_PPLL 功能说明	79
2.9.7 GTP_PPLL 应用模式	85
免责声明	89

表目录

表 1-1 时钟连接概括	12
表 2-1 输入时钟管脚的布局规则	16
表 2-2 GTP_CLKBUFGCE 端口描述	19
表 2-3 GTP_CLKBUFGCE 参数列表描述	19
表 2-4 GTP_CLKBUFGMUX 端口描述	21
表 2-5 GTP_CLKBUFGMUX 参数描述	21
表 2-6 GTP_CLKBUFGMUX_E1 端口描述	23
表 2-7 GTP_CLKBUFGMUX_E1 参数描述	23
表 2-8 GTP_CLKBUFGMUX_E2 端口描述	26
表 2-9 GTP_CLKBUFGMUX_E2 参数描述	27
表 2-10 GTP_CLKBUFGMUX_E2 不同配置时 CLKOUT 输出结果	27
表 2-11 GTP_CLKBUFXCE 端口描述	33
表 2-12 GTP_CLKBUFXCE 参数描述	33
表 2-13 GTP_IOCLKDIV_E2 端口描述	36
表 2-14 GTP_IOCLKDIV_E2 参数描述	36
表 2-15 GTP_IOCLKBUF 端口描述	39
表 2-16 GTP_IOCLKBUF 参数描述	39
表 2-17 GTP_CLKBUFMCE 端口描述	41
表 2-18 GTP_CLKBUFMCE 参数描述	41
表 2-19 GTP_CLKBUFMCE 模式描述	41
表 2-20 GTP_GPLL 端口描述	46
表 2-21 GTP_GPLL 参数描述	48
表 2-22 GTP_GPLL_APB 端口描述	51
表 2-23 GTP_GPLL APB 接口寄存器描述	52
表 2-24 反馈时钟选择	57
表 2-25 输出频率编程参数说明	58
表 2-26 输出时钟相位关系	61
表 2-27 动态相移端口说明	62
表 2-28 GPLL LOCK 指示模式	65
表 2-29 GTP_PPLL 端口列表	72
表 2-30 GTP_PPLL 参数列表	73
表 2-31 GTP_PPLL_APB 接口端口列表	76

表 2-32 GTP_PPLL_APB 接口寄存器列表	77
表 2-33 反馈时钟选择列表	80
表 2-34 PPLL 输出频率编程参数说明	81
表 2-35 输出时钟相位关系	84
表 2-36 PPLL LOCK 指示模式	85

图目录

图 1-1 PG2L100H 时钟资源图.....	10
图 1-2 PG2L100H 时钟架构图.....	11
图 1-3 时钟区域(Clock Region)结构示意图	14
图 2-1 时钟管脚位置示意图	15
图 2-2 时钟管脚连接示意图	16
图 2-3 全局时钟的连接示意图	18
图 2-4 GTP_CLKBUFGCE 端口时序图一.....	20
图 2-5 GTP_CLKBUFGCE 端口时序图二.....	20
图 2-6 CLKBUFGMUX 端口时序图一 (TRIGGER_MODE="NORMAL")	22
图 2-7 CLKBUFGMUX 端口时序图二 (TRIGGER_MODE=" NEGEDGE")	22
图 2-8 CLKBUFGMUX 端口时序图三 (TRIGGER_MODE=" POSEDGE")	22
图 2-9 GTP_CLKBUFGMUX_E1 端口时序图一	24
图 2-10 GTP_CLKBUFGMUX_E1 端口时序图二	24
图 2-11 GTP_CLKBUFGMUX_E2 端口时序图一	28
图 2-12 GTP_CLKBUFGMUX_E2 端口时序图二	29
图 2-13 GTP_CLKBUFGMUX_E2 端口时序图三	29
图 2-14 GTP_CLKBUFGMUX_E2 端口时序图四	30
图 2-15 无 HSSTLP 区域水平时钟相邻区域 HCKB 输入时钟源和输出连接示意图	31
图 2-16 有 HSSTLP 区域水平相邻区域 HCKB 输入时钟源和输出连接示意图	32
图 2-17 GTP_CLKBUFXCE 端口时序图一.....	34
图 2-18 GTP_CLKBUFXCE 端口时序图二.....	34
图 2-19 区域时钟资源与作用域示意图	35
图 2-20 GTP_IOCLKDIV_E2 端口时序图.....	37
图 2-21 IO CLK 时钟资源示意图.....	37
图 2-22 时钟输入时钟源连接示意图.....	38
图 2-23 MRCKB 输入时钟源和输出连接关系示意图.....	40
图 2-24 GTP_CLKBUFMCE 上升沿同步使能输入输出时序图	42
图 2-25 GTP_CLKBUFMCE 下降沿同步使能输入输出时序图	42
图 2-26 GTP_CLKBUFMCE 异步使能模式 0 输入输出时序图	42
图 2-27 GTP_CLKBUFMCE 异步使能模式 1 输入输出时序图	42
图 2-28 PLL 输入时钟源和输出时钟连接示意图	43
图 2-29 GPLL 资源示意图 (Top View)	44

图 2-30 GPLL 系统框图	45
图 2-31 GTP_GPLL 框图	46
图 2-32 GTP_GPLL_APB 写时序	50
图 2-33 GTP_GPLL_APB 读时序	50
图 2-34 GTP_GPLL_APB 写后读时序	51
图 2-35 GTP_GPLL_APB 接口读后写时序	51
图 2-36 GTP_GPLL 参考时钟源	55
图 2-37 GTP_GPLL 复位时序	55
图 2-38 GTP_GPLL 关电时序	56
图 2-39 GTP_GPLL 外部反馈示意图	56
图 2-40 GTP_GPLL 输出频率路径示意图	57
图 2-41 GTP_GPLL 相位调整_静态配置	61
图 2-42 GTP_GPLL 动态相移模式接口时序	62
图 2-43 GPLL 中心扩频示意图	63
图 2-44 GPLL 向下扩频示意图	64
图 2-45 GTP_GPLL OUTPUT Divider 输出级联	64
图 2-46 GTP_GPLL 输出时钟 GATE 时序图	65
图 2-47 GPLL 内部反馈模式示意图	67
图 2-48 GPLL 源同步模式示意图	68
图 2-49 GPLL 零延迟模式示意图	69
图 2-50 PPLL 资源示意图	70
图 2-51 PPLL 系统框图	71
图 2-52 GTP_PPLL 框图	72
图 2-53 GTP_PPLL_APB 写时序	75
图 2-54 GTP_PPLL_APB 读时序	76
图 2-55 GTP_PPLL_APB 写后读时序	76
图 2-56 GTP_PPLL_APB 接口读后写时序	76
图 2-57 GTP_PPLL 参考时钟源	79
图 2-58 GTP_PPLL 复位时序	79
图 2-59 GTP_PPLL 关电时序	80
图 2-60 GTP_PPLL 外部反馈示意图	80
图 2-61 GTP_PPLL 输出频率编程示意图	81
图 2-62 GTP_PPLL 相位调整_静态配置	83
图 2-63 GTP_PPLL 输出时钟 GATE 时序图	85
图 2-64 PPLL 内部反馈模式示意图	86

图 2-65 PPLL 源同步模式示意图.....	87
图 2-66 PPLL 零延迟模式示意图.....	88

1 总体介绍

本文档介绍 Logos2 系列器件时钟资源，以及通用 PLL 的应用和功能特性。其中，时钟资源包括专用时钟管脚资源、时钟网络资源、时钟 BUFFER 资源和时钟 GTP 资源，通用 PLL 包括 GPLL 和 PPLL。

本文中的 GTP 参数默认值详见《UG040007_Logos2 系列产品 GTP 用户指南》。

1.1 时钟资源概述

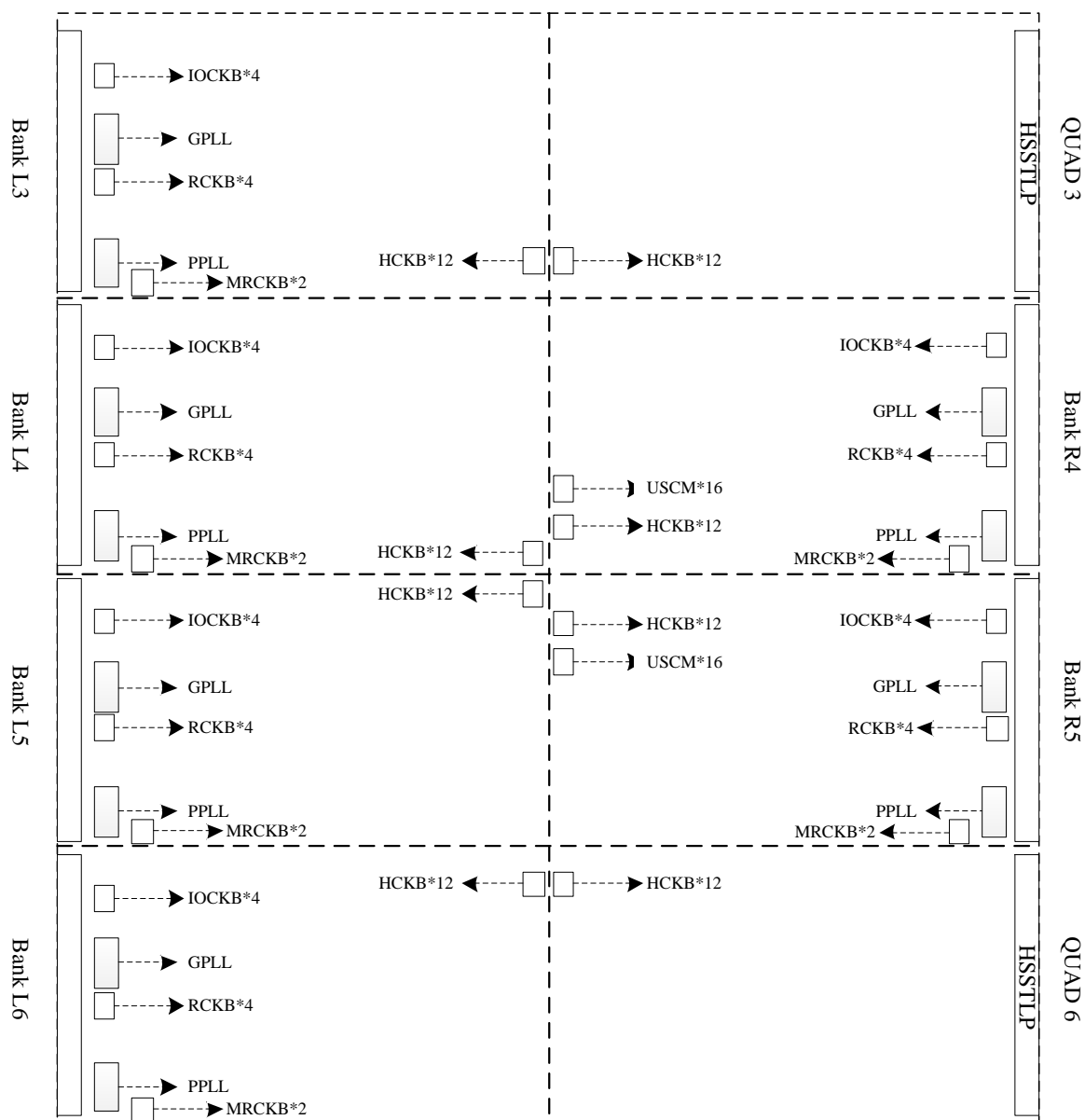


图 1-1 PG2L100H 时钟资源图

Logos2 系列产品提供了丰富的片上时钟资源，示例 PG2L100H 时钟资源如上图所示，包括：32 个 GLOBAL CLK，整个器件被分成上半、下半两个部分，每个部分有各自独立的 16

个 GLOBAL CLK 网络。基于 Region 的 REGIONAL CLK，整个器件被分割成多个 Region，除了 HSSTLP 所在的 Region 外，每个 Region 有 4 个 REGIONAL CLK 网络和 4 个 IO CLK 网络，IOCLK 专用于高速接口应用。另外还有为水平相邻时钟区域提供的 HORIZONTAL CLK 网络以及为垂直相邻时钟区域提供的跨时钟区域时钟网络。

芯片整体时钟资源如下所示：

- PG2L100H 芯片有 32 个 GLOBAL CLOCK，芯片上下部分各有 16 个 GLOBAL CLOCK
- PG2L100H 芯片有 24 个 REGION CLOCK，除了 HSST 所在的两个 REGION 外，其余每个 REGION 有 4 个 REGION CLOCK
- PG2L100H 芯片有 96 个 HORIZONTAL CLOCK，每个 REGION 里有 12 个 HORIZONTAL CLOCK
- PG2L100H 芯片有 24 个 IO CLOCK，每个 IO BANK 有 4 个 IO CLOCK

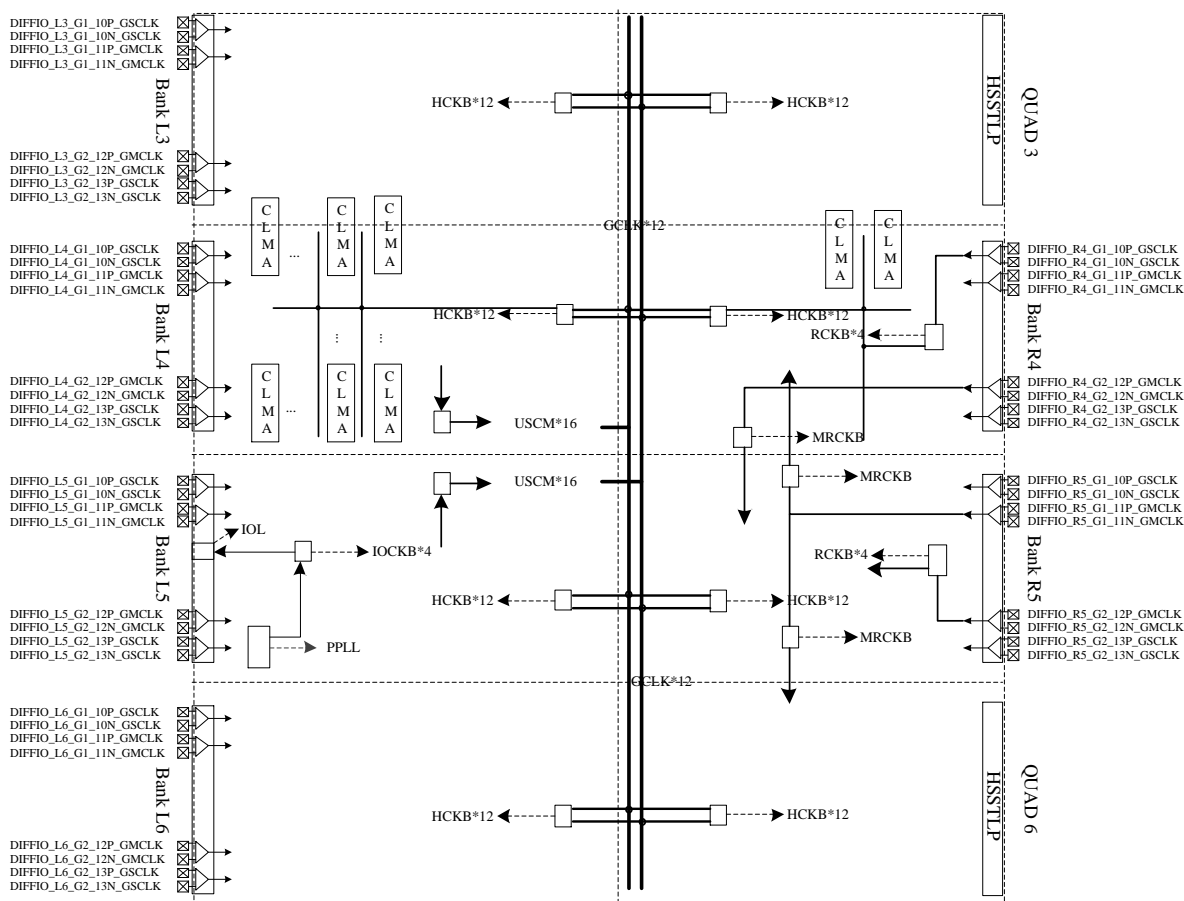


图 1-2 PG2L100H 时钟架构图

上图是 PG2L100H 时钟架构,其它器件时钟架构一样，区域数量和资源数量不一样，由 USCM 纵向驱动的 GLOBAL CLOCK 为全芯片提供同步逻辑单元的同步时钟，其中 16 个 USCM 位于芯片上半区，另外 16 个 USCM 位于芯片下半区。

从上图可以看出，通过时钟管脚可以让用户的时钟到：

- 相同上/下半区的 USCM
- 相同或者相邻水平区域的 HCKB，通过 HCKB 到相同区域的时钟网络
- 相同的区域的 RCKB，通过 RCKB 到区域时钟网络
- 相同区域的 MRCKB（仅支持输入时钟管脚 GMCLK）来为当前时钟区域和相邻的上下时钟区域提供时钟
- 相同区域的 IOCKB，通过 IOCKB 为相同区域的 IOL 提供时钟

1.2 时钟连接概述

下表概括了 Logos2 系列 FPGA 时钟管脚和各个时钟缓冲器的输入时钟来源和输出可直接驱动的资源，明确了相关限制约束，了解所有时钟资源的特性有利于我们更好的做时钟规划和 FPGA 设计。

表 1-1 时钟连接概括

时钟 buffer/管脚	输入时钟源	输出可直接驱动资源
GMCLK,每个 IO bank 有两对差分输入	外部时钟源	相同区域： <ul style="list-style-type: none"> • RCKB • IOCKB • MRCKB 相同区域或相邻水平区域： <ul style="list-style-type: none"> • HCKB 相同左半区或右半区： <ul style="list-style-type: none"> • 一个或者多个 GPLL/PPLL 相同上半区或下半区： <ul style="list-style-type: none"> • USCM
GSCLK,每个 IO bank 有两对差分输入	外部时钟源	相同区域： <ul style="list-style-type: none"> • RCKB • IOCKB 相同区域或相邻水平区域： <ul style="list-style-type: none"> • HCKB 相同左半区或右半区： <ul style="list-style-type: none"> • 一个或者多个 GPLL/PPLL 相同上半区或下半区： <ul style="list-style-type: none"> • USCM
USCM	相同上半区或下半区： <ul style="list-style-type: none"> • GSCLK 和 GMCLK • GPLL 和 PPLL • HCKB（HSSTLP 所在区域不支持） • RCKB • SRB（不推荐） • 相邻 USCM • HSSTLP（该表 HSSTLP 时钟） 	<ul style="list-style-type: none"> • 通过 HCKB 驱动 GPLL/PPLL • 相同半区相邻的 USCM • 任何有全局时钟引脚的逻辑单元 • HSSTLP（该表 HSSTLP 时钟） • 所有 HCKB • CLMA 控制信号
HCKB	相同区域和相邻水平区域： <ul style="list-style-type: none"> • GSCLK 和 GMCLK 	相同区域： <ul style="list-style-type: none"> • GPLL/PPLL

时钟 buffer/管脚	输入时钟源	输出可直接驱动资源
	<ul style="list-style-type: none"> PPLL 和 GPLL HSSTLP (该表 HSSTLP 时钟) 整个芯片: <ul style="list-style-type: none"> 相同上半区/下半区 USCM SRB (不推荐) 	<ul style="list-style-type: none"> HSSTLP (该表 HSSTLP 时钟) 任何有时钟引脚的逻辑单元。 I/O 逻辑 (IOL) 相同上半区/下半区: <ul style="list-style-type: none"> USCM
RCKB	相同区域: <ul style="list-style-type: none"> GSCLK 和 GMCLK PPLL 与 GPLL CLKOUT0~3/N SRB 相同和垂直相邻区域: <ul style="list-style-type: none"> MRCKB 	相同区域: <ul style="list-style-type: none"> GPLL/PPLL 任何有时钟引脚的逻辑单元 <ul style="list-style-type: none"> I/O 逻辑 (IOL) 相同上半区或下半区: <ul style="list-style-type: none"> USCM
IOCKB	相同区域: <ul style="list-style-type: none"> GSCLK 和 GMCLK PPLL 与 GPLL CLKOUT0~3/N 相同和垂直相邻区域: <ul style="list-style-type: none"> MRCKB 	相同区域: <ul style="list-style-type: none"> I/O 逻辑 (IOL)
MRCKB	相同区域: <ul style="list-style-type: none"> GMCLK SRB 	相同和垂直相邻区域: <ul style="list-style-type: none"> RCKB IOCKB
GPLL/PPLL	<ul style="list-style-type: none"> 相同左半区或右半区的 GMCLK 和 GSCLK 相同上半区/下半区 USCM 相同区域的 RCKB 相同区域的 HCKB 相同左侧/右侧 GPLL/PPLL CLKOUT0~3/N 	GPLL: CLKOUT0~3/N、CLKOUT4~6、CLKOUTF/CLKOUTFN: <ul style="list-style-type: none"> 相同半区 USCM 和相同或者相邻水平区域 HCKB CLKOUT0~3: <ul style="list-style-type: none"> 相同区域的 RCKB 和 IOCKB PPLL: CLKOUT0/N、CLKOUT1~4、CLKOUTF/CLKOUTFN: <ul style="list-style-type: none"> 相同半区 USCM 和相同或者相邻水平区域 HCKB CLKOUT0~3: <ul style="list-style-type: none"> 相同区域的 RCKB 和 IOCKB CLKOUTPHY/N: <ul style="list-style-type: none"> DDR PHY
HSSTLP 输入时钟管脚: P_TX_CLK_FR_CORE P_TCLK2_FR_CORE P_RX_CLK_FR_CORE P_RCLK2_FR_CORE	相同上/下半区 USCM 相同区域的 HCKB	无
HSSTLP 时钟输出时钟管脚: P_RCLK2FABRIC P_TCLK2FABRIC	无	<ul style="list-style-type: none"> 相同上/下半区 USCM 相同区域的 HCKB
HSSTLP 参考时钟: REFCLK_CML_P REFCLK_CML_N	外部 HSSTLP 参考时钟	<ul style="list-style-type: none"> 相同上/下半区 USCM 相同区域的 HCKB

注: HSSTLP 时钟详见 《UG040008_Logos2 系列 FPGA 高速串行收发器(HSSTLP)用户指南》

1.3 CLOCK REGION

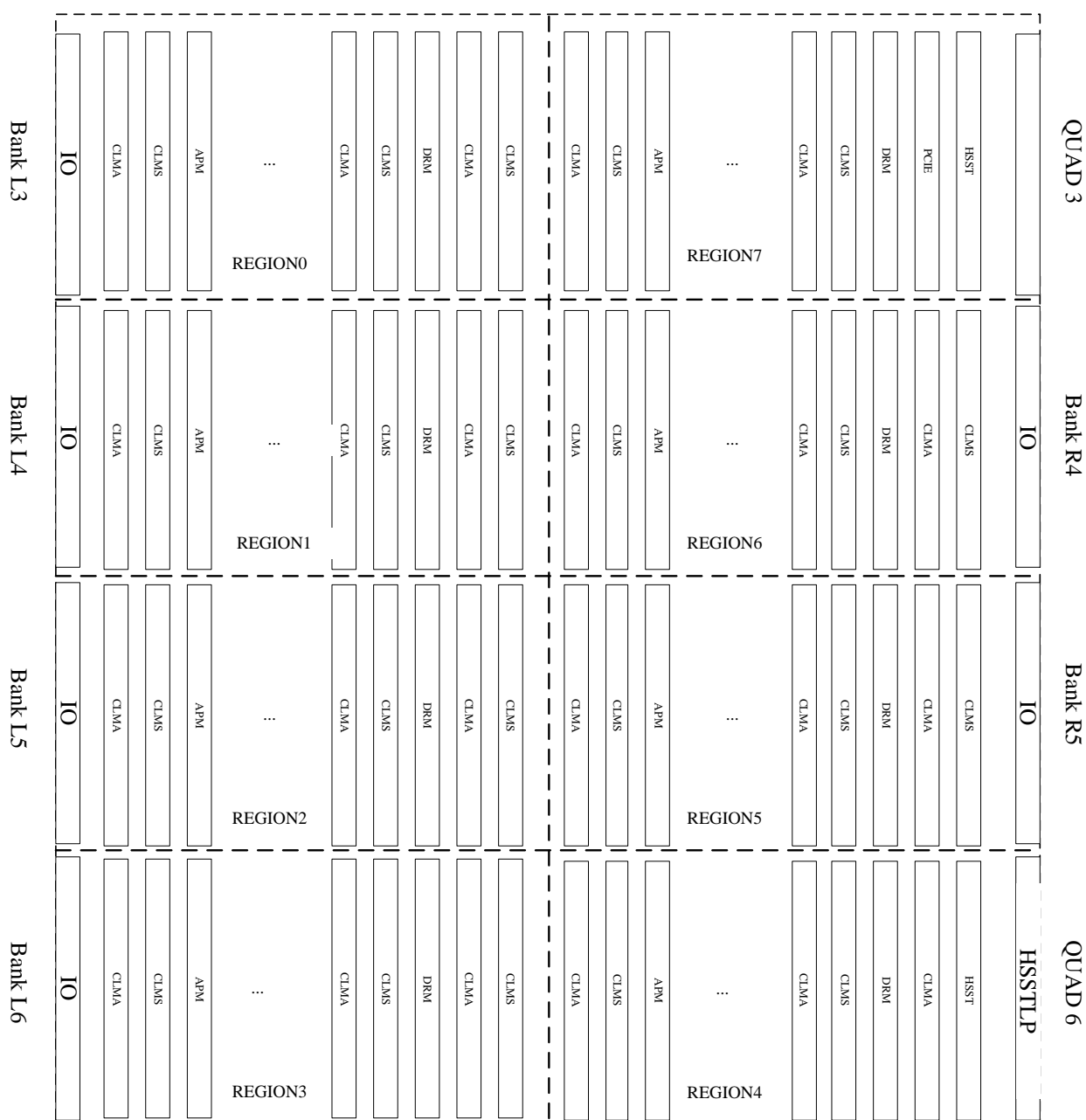


图 1-3 时钟区域(Clock Region)结构示意图

Logos2 系列产品的内部逻辑资源按 Region 分布,每个 Region 由一定数量的 CLMA、CLMS、DRM、IO 及 APM 列等组成,其纵向高度约为 50 个 CLMA 或 CLMS 与 1 个 IO LOGIC 的高度,横向宽度则根据芯片逻辑容量不同而不同,PG2L100H 由 8 个 Region 组成,如上图所示。

2 详细介绍

2.1 时钟管脚资源

2.1.1 时钟管脚介绍

Logos2 系列产品提供具有时钟功能的 IO 管脚 GSCLK/GMCLK，每个 BANK 有两对 GMCLK 和两对 GSCLK 供用户使用。建议用户时钟输入优先选择这些时钟管脚，使用这些管脚可以避免普通布线资源带来的干扰，从而得到更好的时钟性能。这些时钟输入管脚不作为时钟输入时，可作为普通 IO 使用。

时钟 IO 管脚 GSCLK/GMCLK 支持差分或单端使用。当应用于差分信号，需要使用 GSCLK_P/GSCLK_N 或 GMCLK_P/GMCLK_N 差分对端口。当应用于单端信号，则需要使用 GSCLK_P 或 GMCLK_P 单端端口，此时 GSCLK_N/GMCLK_N 不再作为时钟端口使用，只能作为普通 IO 应用，芯片时钟 IO 管脚位置示意图如下所示。

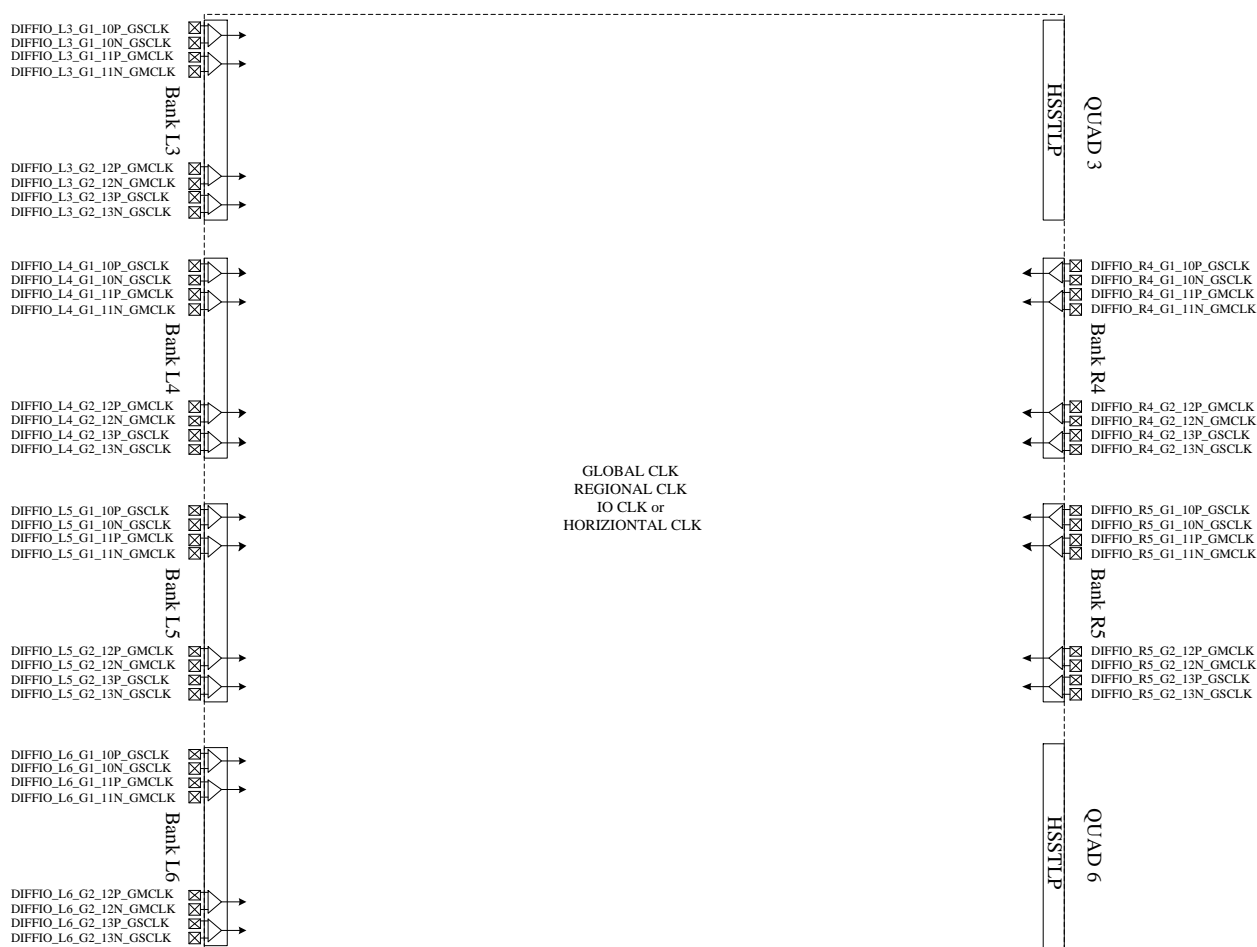


图 2-1 时钟管脚位置示意图

用户需要注意的是 GSCLK 与 GMCLK 时钟输入管脚在使用上是有区别的。体现在

GMCLK 可以通过 MRCKB 作为驱动垂直相邻时钟区域的同步时钟，GSCLK 则不能。如下图所示给出 BankL4 的一对 GSCLK 和一对 GMCLK 的连接关系示意。

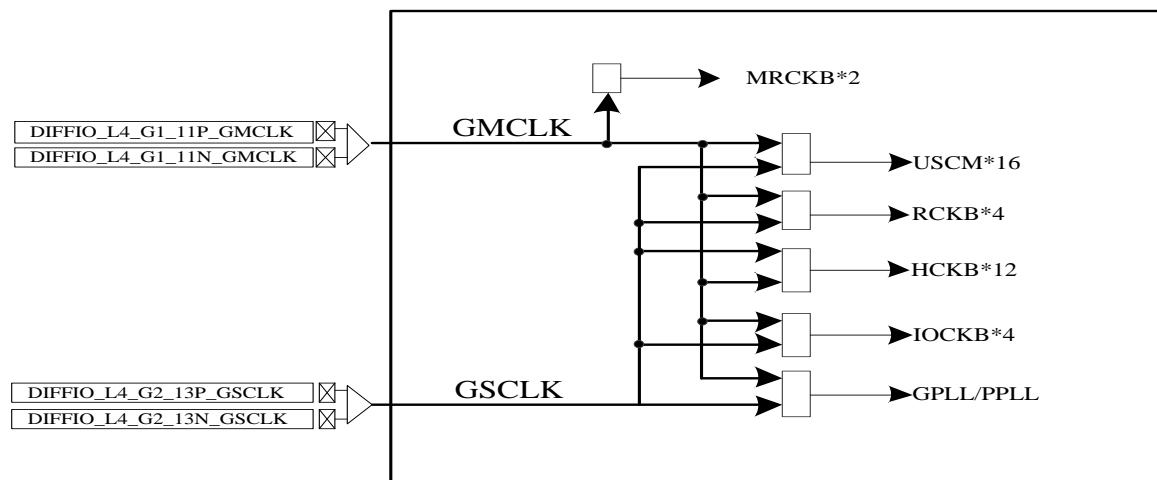


图 2-2 时钟管脚连接示意图

2.1.2 时钟输入管脚布局规则

当选择时钟输入管脚的时候，主要考虑以下几个因素：

- 确保输入时钟管脚能够连接到期望的时钟网络资源
- 确保期望的时钟资源可用并且没有被其它时钟源使用，假如输入时钟期望使用的时钟资源正好被 GPLL 输出时钟占用，这时软件编译就会报错

输入时钟管脚具体布局规则如下表所示：

表 2-1 输入时钟管脚的布局规则

输入时钟到	时钟资源使用和布局规则	有效输入时钟管脚
I/O 资源和整个芯片的逻辑资源	输入时钟管脚→USCM→全局时钟树 <ul style="list-style-type: none"> 时钟管脚必须跟 USCM 在相同的上半区或者下半区 上半区和下半区各有 16 个 USCM 每个区域有 12 个独立全局时钟，与水平时钟共用这些时钟网络 	GMCLK 或 GSCLK
I/O 资源和用 HCKB 驱动单个区域的逻辑资源	输入时钟管脚→HCKB→水平时钟网络 <ul style="list-style-type: none"> 时钟管脚必须放在与 HCKB 相同或者相邻的水平区域 每个区域有 12 个 HCKB 和水平时钟网络 	GMCLK 或 GSCLK
I/O 资源和用 GPLL/PPLL 驱动逻辑资源	整个芯片： <ul style="list-style-type: none"> 输入时钟管脚→GPLL/PPLL→USCM →全局时钟树 单个区域： <ul style="list-style-type: none"> 输入时钟管脚→GPLL/PPLL →RCKB→区域时钟树 单个或者水平相邻区域 <ul style="list-style-type: none"> 时钟管脚输入→GPLL/PPLL→HCKB→水平时钟网络 	GMCLK 或 GSCLK

输入时钟到	时钟资源使用和布局规则	有效输入时钟管脚
	输入时钟到 GPLL/PPLL 布线规则： <ul style="list-style-type: none"> • 相同左半区或者右半区之间通过 PLL 时钟树 • 跨左右半区需要通过 USCM 或相邻水平区域的 HCKB 到 GPLL/PPLL • HSSTLP 所在区域外，每个区域各有 1 一个 GPLL/PPLL 	
I/O 资源和用 RCKB 驱动单个区域逻辑资源	输入时钟管脚→RCKB→区域时钟树 <ul style="list-style-type: none"> • 输入时钟管脚必须与 RCKB，I/O 资源，和被该时钟驱动的逻辑资源在相同区域 • HSSTLP 所在区域外，每个区域有 4 对差分时钟管脚输入和 4 个 RCKB 	GMCLK 或 GSCLK
I/O 资源和用 MRCKB 驱动相邻三个区域逻辑资源	输入时钟管脚→MRCKB→RCKB→区域时钟树 <ul style="list-style-type: none"> • RCKB 所驱动的 I/O 资源和逻辑资源必须与输入时钟管脚在相同或者相邻垂直水平区域 • MRCKB 所驱动的 RCKB 必须与输入时钟管脚在相同或者相邻垂直水平区域 • HSSTLP 所在区域外，每个区域有 4 对差分时钟管脚输入、4 个 RCKB 和 2 个 MRCKB 	仅 GMCLK
同一个区域的高速接口	输入时钟管脚→IOCKB→I/O 时钟树 <ul style="list-style-type: none"> • 输入时钟管脚必须与 IOCKB 和它所驱动的 IOL 在相同区域 • HSSTLP 所在区域外，每个区域有 4 对差分时钟管脚输入、4 个 IOCKB 和 2 个 MRCKB 	GMCLK 或 GSCLK
相同和相邻垂直区域的高速接口	输入时钟管脚→MRCKB→IOCKB→I/O 时钟树 <ul style="list-style-type: none"> • 输入时钟管脚必须与 IOCKB 和它所驱动的 IOL 在相同或者相邻垂直区域 • MRCKB 所驱动的 IOCKB 必须与输入时钟管脚在相同或者相邻垂直水平区域 • HSSTLP 所在区域外，每个区域有 4 对差分时钟管脚输入、4 个 IOCKB 和 2 个 MRCKB 	仅 GMCLK

2.2 全局时钟资源

2.2.1 全局时钟网络

USCM 作为全局时钟的驱动器，为时钟区域内的同步逻辑单元提供全局时钟 GCLK。全局时钟来源于 GSCLK_P、GMCLK_P、PPLL、GPLL、HSST、HCKB、RCKB、SRB 等的时钟。USCM 的连接关系如下虚线圈所示。

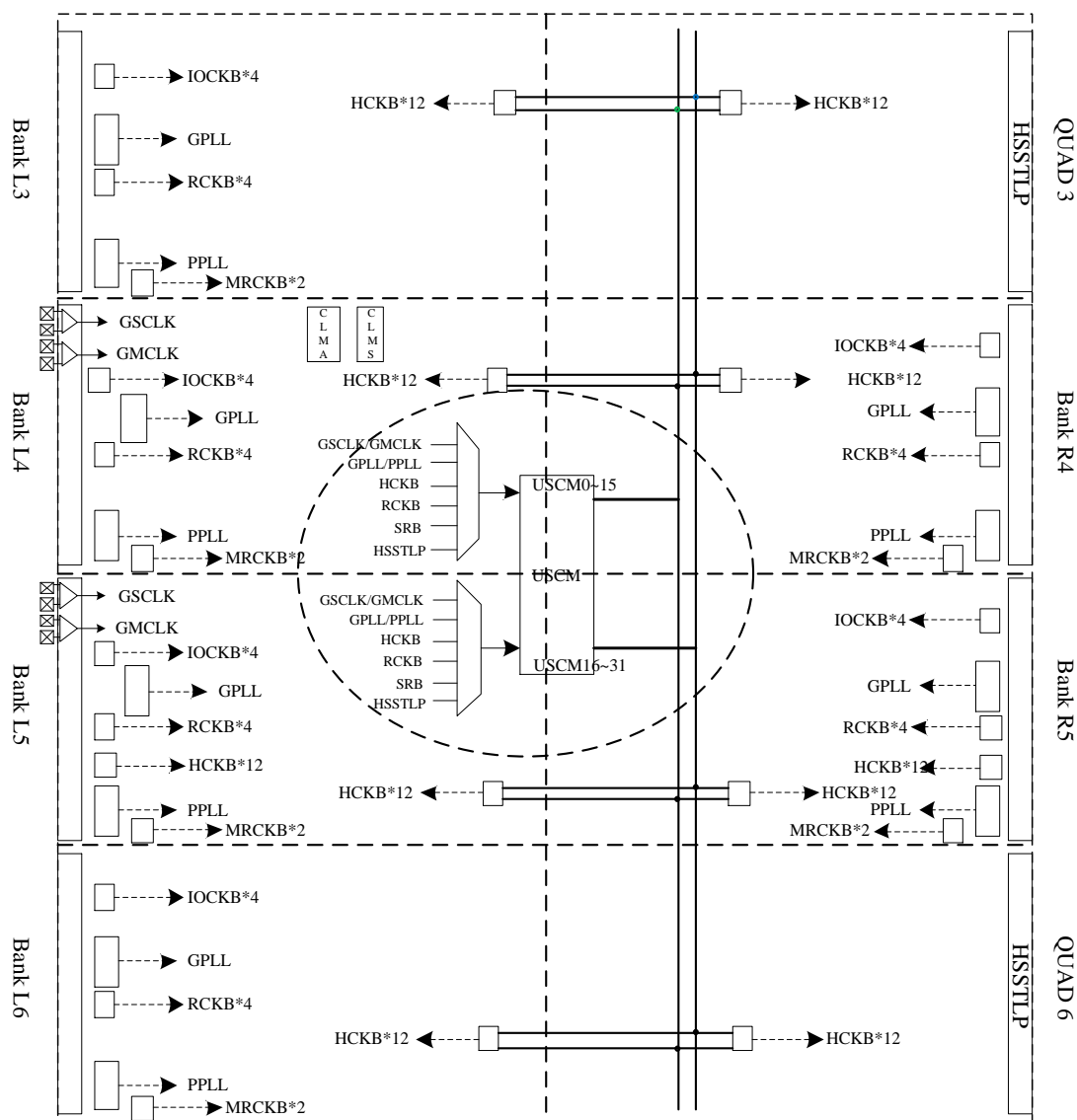


图 2-3 全局时钟的连接示意图

如上图所示，虚线圈横向虚线将芯片分成上半区和下半区，纵向虚线将芯片分成左半区和右半区。位于上半区的时钟输入源只能到达上半区的 16 个 USCM，位于下半区的时钟输入源只能到达下半区的 16 个 USCM，HSSTLP 占用的时钟资源较多，且只能使用全局时钟资源和水平时钟资源，优先考虑使用水平时钟资源。

全局时钟的时钟源 GMCLK/GSCLK,GPLL/PPLL,HCKB,RCKB,分布在 6 个时钟区域,全

局时钟的时钟源 HSSTLP 分布在 2 个时钟区域。每个时钟区域最多可同时提供 14 个时钟源上全局时钟。用户在使用全局时钟时需要考虑到这一点，合理布局时钟源的位置。另外每个时钟区域至多有 12 个全局时钟驱动逻辑。

2.2.2 全局时钟 GTP

USCM 为 GLOBAL CLK 提供了时钟门控和时钟选择的功能，它支持 CLKBUFG，CLKBUFGCE 以及 CLKBUFGMUX 三种模式。

CLKBUFG 提供了简单的时钟 Buffer 功能，通常由软件自动插入。CLKBUFGCE 属于带时钟门控功能的时钟 BUFFER，它的使用通过直接在 RTL 中例化 GTP 来实现。CLKBUFGMUX 可实现在两个时钟源之间的动态切换，也需要通过在代码中例化 GTP 来实现。

2.2.2.1 GTP_CLKBUFGCE

GTP_CLKBUFGCE 原语例化：

```
GTP_CLKBUFGCE #(
.DEFAULT_VALUE    (1'b0      ),
.SIM_DEVICE       ("LOGOS2"   )
) GTP_CLKBUFGCE_INST (
.CLKOUT           (CLKOUT     ),
.CE               (CE         ),
.CLKIN            (CLKIN      )
);
```

表 2-2 GTP_CLKBUFGCE 端口描述

端口信号	输入/输出	说明
CLKIN	输入	输入时钟
CE	输入	时钟使能，高有效使能
CLKOUT	输出	输出时钟

表 2-3 GTP_CLKBUFGCE 参数列表描述

参数名称	参数类型	设置值	功能描述
DEFAULT_VALUE	<string>	1'b0, 1'b1	CE=0 时，clkout 输出 DEFAULT_VALUE； CE=1 时，clkout 输出 CLKIN；
SIM_DEVICE	<string>	“TITAN” “LOGOS” “COMPACT” “LOGOS2”	仿真模型器件识别 “TITAN”：在上升沿或者下降沿触发时，时钟打 2 拍就切换 “LOGOS” “COMPACT” “LOGOS2” “TITAN2”：

参数名称	参数类型	设置值	功能描述
			在上升沿或者下降沿触发时，时钟打 4 拍才切换

GTP 时序

参数DEFAULT_VALUE=1时，使用CLKIN上升沿采样，时序如下图所示。当CE保持高电平时，输入时钟信号被传输到输出时钟端口；当CE从高电平变成低电平并保持不变时，CLKIN采样两个CE的低电平后输出时钟保持高电平；当CE从低电平变成高电平并保持不变时，CLKIN采样两个CE的高电平后输出时钟开始翻转。

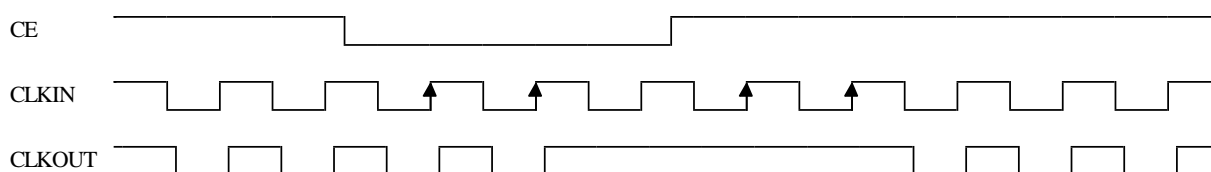


图 2-4 GTP_CLKBUFGCE 端口时序图一

参数DEFAULT_VALUE=0时，使用CLKIN下降沿采样，时序如下图所示。当CE保持高电平时，输入时钟信号被传输到输出时钟端口；当CE从高电平变成低电平并保持低电平不变时，CLKIN采样两个CE的低电平后输出时钟保持低电平；当CE从低电平变成高电平并保持高电平不变时，CLKIN采样两个CE的高电平后输出时钟开始翻转。

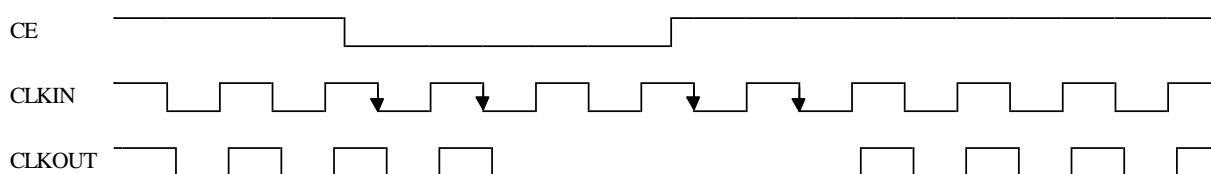


图 2-5 GTP_CLKBUFGCE 端口时序图二

2.2.2.2 GTP_CLKBUFGMUX

还可以例化带时钟选择功能的 GTP_CLKBUFGMUX 单元，GTP_CLKBUFGMUX 可实现在两个时钟源之间的动态切换，用户可以根据需要选择强制切换（对应 TRIGGER_MODE="NORMAL"）；由时钟下降沿触发的 glitchless 切换（对应

TRIGGER_MODE=" NEGEDGE "); 以及由时钟上升沿触发的 glitchless 切换 (对应 TRIGGER_MODE=" POSEDGE ")。

用户在使用 TRIGGER_MODE="NORMAL"切换输入时钟时会引入 glitch，切换后需对相关逻辑复位，以实现预期功能。

CLKBUFGMUX 原语例化：

```
GTP_CLKBUFGMUX #(
    .TRIGGER_MODE ("NEGEDGE" ),
    .SIM_DEVICE    ("LOGOS2" )
) GTP_CLKBUFGMUX_INST (
    .CLKOUT        (CLKOUT      ),
    .CLKIN0        (CLKIN0      ),
    .CLKIN1        (CLKIN1      ),
    .SEL           (SEL         )
);
```

表 2-4 GTP_CLKBUFGMUX 端口描述

端口信号	输入/输出	说明
CLKIN0	输入	输入时钟 0
CLKIN1	输入	输入时钟 1
SEL	输入	时钟选择信号，0 选择 CLK0；1 选择 CLK1；
CLKOUT	输出	输出时钟

表 2-5 GTP_CLKBUFGMUX 参数描述

参数名称	参数类型	有效值	功能描述
TRIGGER_MODE	<string>	“NORMAL” “NEGEDGE” “POSEDGE”	(1) “NORMAL”：此模式下，自由切换输入时钟（CLKIN0和CLKIN1），不提供去毛刺功能 (2) “NEGEDGE”：此模式下，自由切换输入时钟（CLKIN0和CLKIN1），提供去毛刺功能，由时钟下降沿触发。 (3) “POSEDGE”：此模式下，自由切换输入时钟（CLKIN0和CLKIN1），提供去毛刺功能，由时钟上升沿触发。
SIM_DEVICE	<string>	“TITAN” “LOGOS” “COMPACT” “LOGOS2”	仿真模型器件识别 “TITAN”：在上升沿或者下降沿触发时，时钟打2拍就切换 “LOGOS” “COMPACT” “LOGOS2” “TITAN2”：在上升沿或者下降沿触发时，时钟打4拍才切换

GTP 时序

GTP_CLKBUFGMUX 时序图如下所示，当分别在 `TRIGGER_MODE="NEGEDGE"` 和 `TRIGGER_MODE="POSEDGE"` 模式下进行时钟切换时，其电平分别保持当前值到下一个时钟下降沿或上升沿。

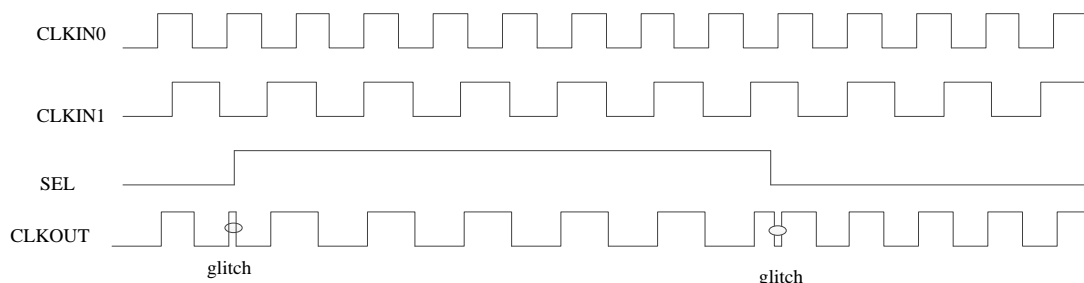


图 2-6 CLKBUFFGMUX 端口时序图一（`TRIGGER_MODE="NORMAL"`）

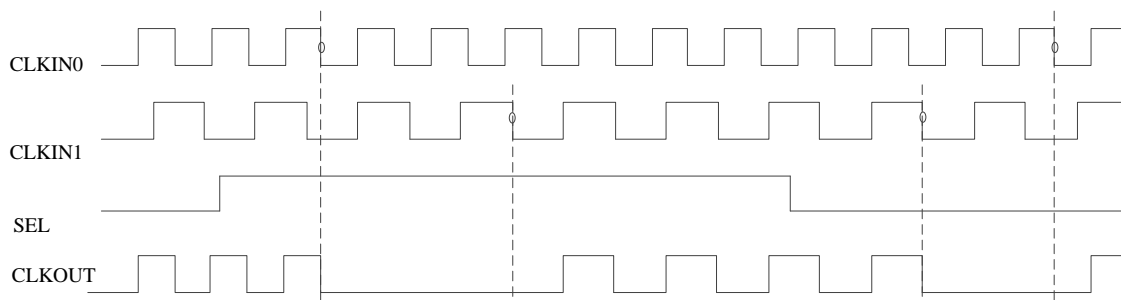


图 2-7 CLKBUFFGMUX 端口时序图二（`TRIGGER_MODE="NEGEDGE"`）

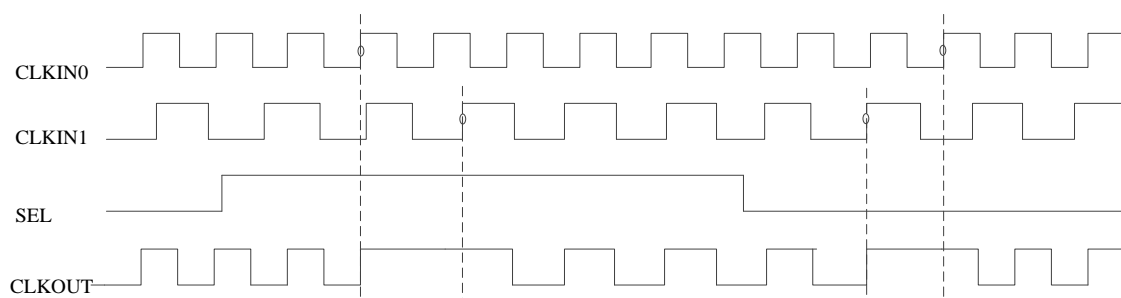


图 2-8 CLKBUFFGMUX 端口时序图三（`TRIGGER_MODE="POSEDGE"`）

2.2.2.3 GTP_CLKBUFGMUX_E1

GTP_CLKBUFGMUX_E1 具有 GTP_CLKBUFGCE 和 GTP_CLKBUFGMUX 两种 GTP 的部分功能，通过 `EN` 信号控制 `CLKOUT` 是否输出时钟信号，通过 `SEL` 信号实现输入时钟动态切换的功能，必须保证当前切换动作完成才能进行下一次切换；与 GTP_CLKBUFGMUX 的参数设置不一样，只支持两种切换模式，"`NEGEDGE`" 模式和 "`POSEDGE`" 模式，参数"

INIT_SEL"可以设置初始输出时钟。

GTP_CLKBUFGMUX_E1 原语例化：

```

GTP_CLKBUFGMUX_E1 #(
    .TRIGGER_MODE ("NEGEDGE" ),
    .INIT_SEL      ("CLK0"      )
) GTP_CLKBUFGMUX_E1_INST(
    .CLKOUT        (CLKOUT      ),
    .CLKIN0        (CLKIN0      ),
    .CLKIN1        (CLKIN1      ),
    .EN            (EN          ),
    .SEL           (SEL          ) );
    
```

表 2-6 GTP_CLKBUFGMUX_E1 端口描述

端口信号	输入/输出	说明
CLKIN0	输入	输入时钟 0
CLKIN1	输入	输入时钟 1
EN	输入	时钟使能控制信号： EN = 1, CLKOUT 输出时钟信号 EN = 0, CLKOUT 不输出时钟信号
SEL	输入	时钟选择控制信号： SEL = 1, 选择输出 CLKIN1 SEL = 0, 选择输出 CLKIN0
CLKOUT	输出	输出时钟

表 2-7 GTP_CLKBUFGMUX_E1 参数描述

参数名称	参数类型	有效值	功能描述
TRIGGER_MODE	<string>	“POSEDGE” “NEGEDGE”	选择时钟上升沿触发还是下降沿触发： 当设置为 “NEGEDGE”时，采用下降沿触发； 当设置为 “POSEDGE”时，采用上升沿触发。
INIT_SEL	<string>	“CLK0” “CLK1”	选择初始输出时钟： 当设置为 “CLK0”时，CLKOUT = CLKIN0； 当设置为 “CLK1”时，CLKOUT = CLKIN1。

GTP 时序

当TRIGGER_MODE = “NEGEDGE”，INIT_SEL = “CLK0”，下降沿触发，初始时钟选择 CLKIN0时，端口时序图如下图：

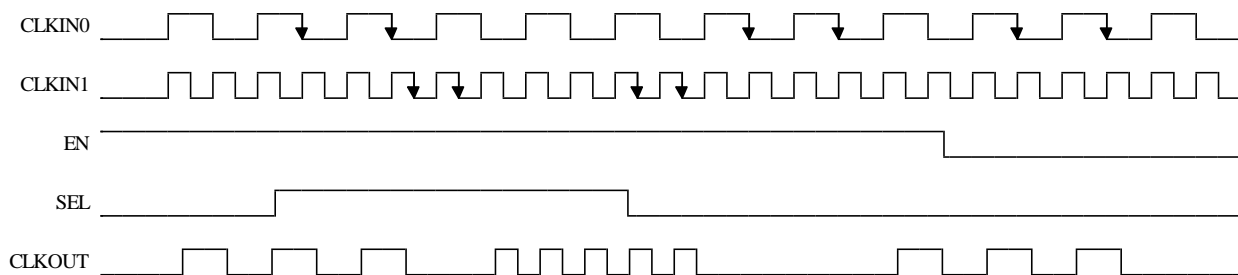


图 2-9 GTP_CLKBUFGMUX_E1 端口时序图一

1. 当EN = 1时

- SEL 从 0 变化到 1，需要先检测当前时钟 CLKIN0 的下降沿 2 次，再检测被切换时钟 CLKIN1 的下降沿 2 次（在检测被切换时钟下降沿的过程中，CLKOUT 输出低电平）。检测完成后 CLKOUT 由 CLKIN0 切换到 CLKIN1
- 当 SEL 从 1 变化到 0 时，先检测 CLKIN1 的下降沿 2 次，再检测 CLKIN0 的下降沿 2 次（检测被切换时钟 CLKIN0 下降沿的过程中，CLKOUT 输出低电平）。检测完成后 CLKOUT 由 CLKIN1 切换到 CLKIN0

2. 当EN = 0时，不允许SEL信号有跳变。上图中EN由1切换到0时检测2次CLKIN0的下降沿后，输出低电平。

当TRIGGER_MODE = “POSEDGE”，INIT_SEL = “CLK0”，上升沿触发，初始时钟选择 CLKIN0时端口时序如下图：

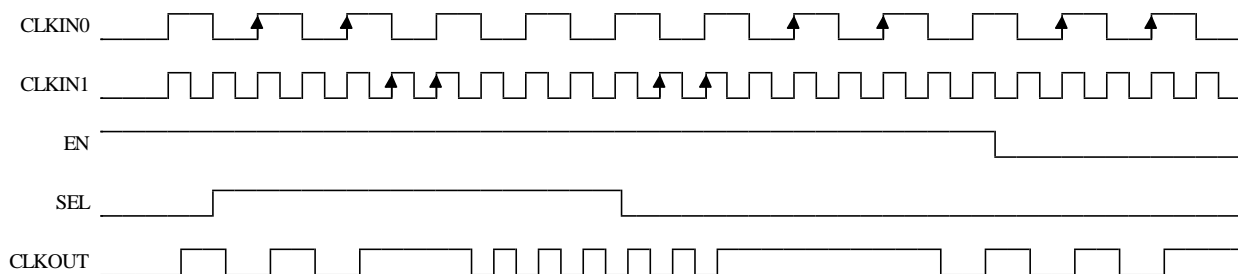


图 2-10 GTP_CLKBUFGMUX_E1 端口时序图二

1. 当EN = 1时

- SEL 从 0 变化到 1，需要先检测当前时钟 CLKIN0 的上升沿 2 次，再检测被切换时钟 CLKIN1 的上升沿 2 次（在检测被切换时钟上升沿的过程中，CLKOUT 输出高电平）。检测完成后 CLKOUT 由 CLKIN0 切换到 CLKIN1
- 当 SEL 从 1 变化到 0 时，先检测 CLKIN1 的上升沿 2 次，再检测 CLKIN0 的上升沿 2 次（检测被切换时钟 CLKIN0 上升沿的过程中，CLKOUT 输出高电平）。检测完成后 CLKOUT 由 CLKIN1 切换到 CLKIN0

2. 当EN = 0时，不允许SEL信号有跳变。EN由1切换到0时检测2次CLKOUT的上升沿后，输出高电平。

当 EN=1,只有一个时钟源有效时，需要保证 INIT_SEL 参数设置与有效时钟源一致，且 SEL 选择的是有效时钟输出：

- CLKIN0 有效，CLKIN1=1/0，INIT_SEL=“CLK0”，SEL=0，CLKOUT=CLKIN0
- CLKIN0 有效，CLKIN1=1/0，INIT_SEL=“CLK0”，SEL=1，CLKOUT=0/1
- CLKIN0=1/0，CLKIN1 有效，INIT_SEL=“CLK1”，SEL=0，CLKOUT=0/1
- CLKIN0=1/0，CLKIN1 有效，INIT_SEL=“CLK1”，SEL=1，CLKOUT= CLKIN1

TRIGGER_MODE 设置为“POSEDGE”，CLKOUT 输出保持不变时值为 1，设置为“NEGEDGE”，CLKOUT 输出保持不变时值为 0。

注意：

INIT_SEL 设置的初始输出时钟和 SEL 信号与有效时钟源不一致时，无法保证 CLKOUT 输出有效时钟。例如输入时钟 CLKIN0 正常翻转，输入时钟 CLKIN1=1/0，INIT_SEL 配置为“CLK0”初始输出时钟选择 CLKIN0，SEL 设置为 1 选择 CLKIN1 输出时，下载位流后输出时钟保持低电平，即使将 SEL 设置为 0 选择 CLKIN0 输出，输出时钟依旧保持低电平。同理，如果起初两路时钟源有效，中途某路时钟源丢失，也可能出现相同问题。当两路时钟源都有效时恢复正常工作。

2.2.2.4 GTP_CLKBUFGMUX_E2

GTP_CLKBUFGMUX_E2 也有输入时钟动态切换的功能，具有 GTP_CLKBUFGMUX 和 GTP_CLKBUFGMUX_E1 的部分功能特性，与这两个 GTP 最大不同的是该 GTP 可以选择对哪个输入时钟沿检测进行切换，当选择两路时钟沿检测、其中一路时钟沿检测，或两路时钟沿都不检测时,都必须保证当前切换动作完成才能进行下一次切换；当只检测其中一路输入时

钟沿的时候，SEL 信号可以被看做一个使能信号，例如当 DETECT_CLK0=1、DETECT_CLK1=0 时，SEL 为 0 输出时钟与 CLKIN0 一致，SEL 为 1 输出时钟保持 1/0 不变，当 DETECT_CLK0=0、DETECT_CLK1=1 时，SEL 为 0 输出时钟保持 1/0 不变，SEL 为 1 输出时钟与 CLKIN1 一致。根据实际应用需求来选择这 3 个 GTP。

需要注意的以下几点：

- 两路输入时钟沿都不检测，输入时钟切换时会有毛刺
- 如果开始设置两路输入时钟沿都检测，中途改变只使用其中一路检测，在 DETECT_CLK0 和 DETECT_CLK0=1 从 1 跳变到 0 是也可能出现毛刺，参考图 2-12。
- 双时钟沿检测切换需要等待 4 拍、单时钟沿检测切换需要等到 2 拍、都不检测立刻切换，具体查看接口时序描述

GTP_CLKBUFGMUX_E2 原语例化：

```
GTP_CLKBUFGMUX_E2 #(
    .TRIGGER_MODE ("NEGEDGE"      ),
    .INIT_SEL      ("CLK0"        )
) GTP_CLKBUFGMUX_E2_INST (
    .CLKOUT        (CLKOUT        ),
    .CLKIN0        (CLKIN0        ),
    .CLKIN1        (CLKIN1        ),
    .DETECT_CLK0   (DETECT_CLK0   ),
    .DETECT_CLK1   (DETECT_CLK1   ),
    .SEL           (SEL           ));
```

表 2-8 GTP_CLKBUFGMUX_E2 端口描述

端口信号	输入/输出	说明
CLKIN0	输入	输入时钟 0
CLKIN1	输入	输入时钟 1
DETECT_CLK0	输入	设置为 1 时：在切换时钟的过程中，检测 CLKIN0 的时钟沿 设置为 0 时：在切换时钟的过程中，不检测 CLKIN0 的时钟沿
DETECT_CLK1	输入	设置为 1 时：在切换时钟的过程中，检测 CLKIN1 的时钟沿 设置为 0 时：在切换时钟的过程中，不检测 CLKIN1 的时钟沿
SEL	输入	时钟选择控制信号 SEL = 1，选择 CLKIN1 SEL = 0，选择 CLKIN0
CLKOUT	输出	输出时钟

表 2-9 GTP_CLKBUFGMUX_E2 参数描述

参数名称	参数类型	有效值	功能描述
TRIGGER_MODE	<string>	“POSEDGE” “NEGEDGE”	选择时钟上升沿触发还是下降沿触发： 当设置为 “NEGEDGE”时，采用下降沿触发 当设置为 “POSEDGE”时，采用上升沿触发
INIT_SEL	<string>	“CLK0” “CLK1”	选择初始输出时钟： 当设置为 “CLK0”时，CLKOUT = CLKIN0 当设置为 “CLK1”时，CLKOUT = CLKIN1

DETECT_CLK0、DETECT_CLK1、SEL 和 TRIGGER_MODE 不同配置情况 CLKOUT 输出下表所示，表中 CLKIN0 有效，CLKIN1=1/0 时，INIT_SEL 参数设置为“CLK0”；CLKIN0=1/0，CLKIN1 有效时，INIT_SEL 参数设置为“CLK1”。只有一个时钟源有效时，需要保证 INIT_SEL 参数设置与有效时钟源一致。

表 2-10 GTP_CLKBUFGMUX_E2 不同配置时 CLKOUT 输出结果

CLKIN0	CLKIN1	DETECT_CLK0	DETECT_CLK1	SEL	CLKOUT
有效	有效	0	0	0	CLKOUT=CLKIN0
		0	0	1	CLKOUT=CLKIN1
		1	0	0	CLKOUT=CLKIN0
		1	0	1	"NEGEDGE": 0 "POSEDGE": 1
		0	1	0	"NEGEDGE": 0 "POSEDGE": 1
		0	1	1	CLKOUT=CLKIN1
		1	1	0	CLKOUT=CLKIN0
		1	1	1	CLKOUT=CLKIN1
有效	1/0	0	0	0	CLKOUT=CLKIN0
		0	0	1	0
		1	0	0	CLKOUT=CLKIN0
		1	0	1	"NEGEDGE": 0 "POSEDGE": 1
		0	1	0	"NEGEDGE": 0 "POSEDGE": 1
		0	1	1	"NEGEDGE": 0 "POSEDGE": 1
		1	1	0	CLKOUT=CLKIN0
		1	1	1	"NEGEDGE": 0 "POSEDGE": 1
1/0	有效	0	0	0	0
		0	0	1	CLKOUT=CLKIN1
		1	0	0	"NEGEDGE": 0 "POSEDGE": 1
		1	0	1	"NEGEDGE": 0 "POSEDGE": 1

CLKIN0	CLKIN1	DETECT_CLK0	DETECT_CLK1	SEL	CLKOUT
		0	1	0	"NEGEDGE": 0 "POSEDGE": 1
		0	1	1	CLKOUT=CLKIN1
		1	1	0	"NEGEDGE": 0 "POSEDGE": 1
		1	1	1	CLKOUT=CLKIN1

GTP 时序

下图中的 CLKOUT_NEG 和 CLKOUT_POS 分别指参数 TRIGGER_MODE 设置为“NEGEDGE”和“POSEDGE”时的 CLKOUT 输出。这样设置是为了对照方便，并不是说这个 GTP 有 2 个时钟输出端。1 个 GTP 只有 1 个输出端 CLKOUT。

当 DETECT_CLK0 = 1 和 DETECT_CLK1 =1 时，都不忽略对两个时钟沿的检测。CLKOUT_NEG 的初始时钟为 CLKIN0,端口时序如下图所示：

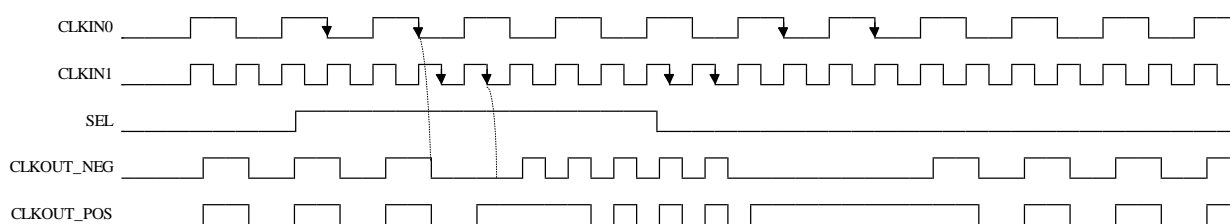


图 2-11 GTP_CLKBUFGMUX_E2 端口时序图一

以CLKOUT_NEG为例：

1. 当SEL从0变化到1时，需要先检测当前时钟CLKIN0的下降沿2次，再检测被切换时钟CLKIN1的下降沿2次（检测CLKIN1下降沿2次的期间内，CLKOUT_NEG输出低电平）。检测完成后，CLKOUT_NEG才由CLKIN0切换到CLKIN1。

2. 当SEL从1变化到0时，需要先检测当前时钟CLKIN1的下降沿2次，再检测被切换时钟CLKIN0的下降沿2次（检测CLKIN0下降沿2次的期间内，CLKOUT_NEG输出低电平）。检测完成后，CLKOUT_NEG才由CLKIN1切换到CLKIN0。

CLKOUT_POS切换过程与CLKOUT_NEG类似，只是把检测下降沿换成检测上升沿。检测被切换时钟的上升沿的时候，CLKOUT_POS切换过程后两拍输出高电平。

当DETECT_CLK0=1, DETECT_CLK1由1切换到0时, 从检测CLKIN0和CLKIN1的时钟沿变成只检测CLKIN0的时钟沿变化, 忽略CLKIN1的时钟沿检测, 端口时序如下图所示:

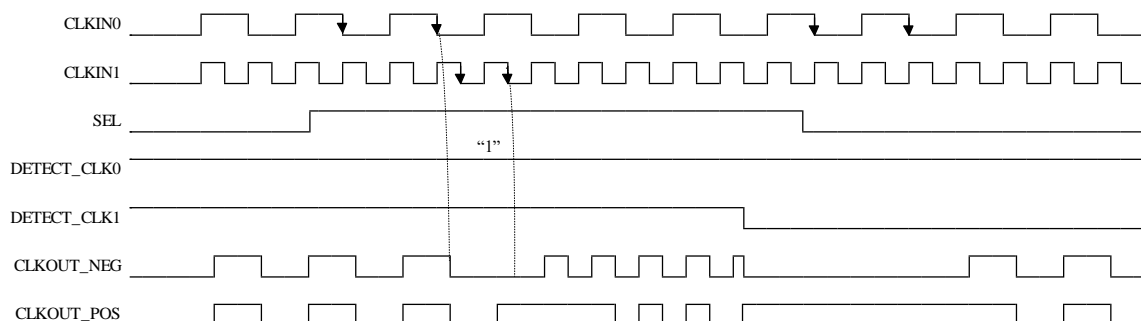


图 2-12 GTP_CLKBUFGMUX_E2 端口时序图二

以 CLKOUT_NEG 为例, 初始时钟为 CLKIN0

1. 当设置了DETECT_CLK1 = 0后, CLKOUT_NEG不再输出CLKIN1, 而是输出低电平。此时SEL从1变化到0时, 不再检测CLKIN1的时钟沿, 直接检测两次CLKIN0的下降沿 (检测CLKIN0下降沿期间CLKOUT_NEG输出低电平)。检测完成后, CLKOUT_NEG切换成CLKIN0。

2. CLKOUT_POS的切换过程与CLKOUT_NEG类似, 只是把检测下降沿换成检测上升沿。检测被切换时钟的上升沿的时候, CLKOUT_POS输出高电平。

当DETECT_CLK0 = 0, DETECT_CLK1 = 1时, 忽略CLKIN0的时钟沿变化, 检测CLKIN1的时钟沿变化, 端口时序如下图:

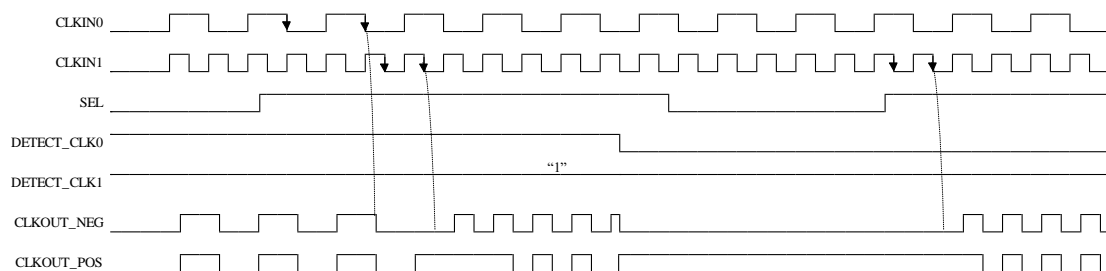


图 2-13 GTP_CLKBUFGMUX_E2 端口时序图三

以 CLKOUT_NEG 为例, 初始时钟为 CLKIN1:

当设置了DETECT_CLK0 = 0后, CLKOUT_NEG不再输出CLKIN0, 而是输出低电平。此时SEL从0变化到1时, 不再检测CLKIN0的时钟沿, 直接检测两次CLKIN1的下降沿 (检测CLKIN1下降沿期间CLKOUT_NEG输出低电平)。检测完成后, CLKOUT_NEG切换成CLKIN1。

CLKOUT_POS的切换过程与CLKOUT_NEG类似, 只是把检测下降沿换成检测上升沿。

检测被切换时钟的上升沿的时候，CLKOUT_POS输出高电平。

当DETECT_CLK0 = 0，DETECT_CLK1 = 0时这种情况下都忽略CLKIN0和CLKIN1的时钟沿检测，时钟切换时不检测当前时钟和被切换时钟的时钟沿变化，而是当SEL有变化的时候，CLKOUT_NEG和CLKOUT_POS立即切换时钟。在这个模式下，输出时钟可能会产生毛刺，接口时序如下图所示：

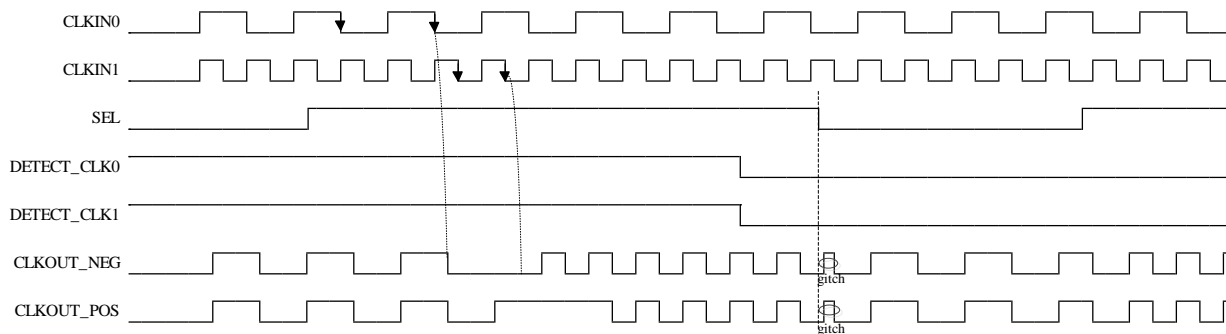


图 2-14 GTP_CLKBUFGMUX_E2 端口时序图四

2.3 水平时钟资源

2.3.1 水平时钟网络

HCKB 是时钟区域内的全局时钟水平缓冲器，负责驱动水平时钟网络。每个 REGION 内都有 12 个 HCKB。

HCKB 的时钟输入源来自水平方向的两个 REGION 内的时钟，包括 GSCLK、GMCLK、PPLL 输出时钟、GPLL 输出时钟、SRB、HSSTLP 输出时钟、USCM 输出时钟（全局时钟）等，HCKB 提供同步动态使能或异步动态使能的功能。

时钟资源对称的相邻水平区域的 HCKB 时钟输入来源的连接关系与图 2-15 一致；相邻水平区域有 HSSTLP 和无 HSSTLP 的区域的 HCKB 时钟输入来源的连接关系与图 2-16 一致。图示中 HCKB 只能到达相同上半区或下半区的 USCM，HSSTLP 所在区域的 HCKB 输出不能作为 USCM 时钟源，而所有 USCM 的输出可以到达芯片所有 HCKB，为了保证时钟信号质量，不建议使用 SRB 时钟。

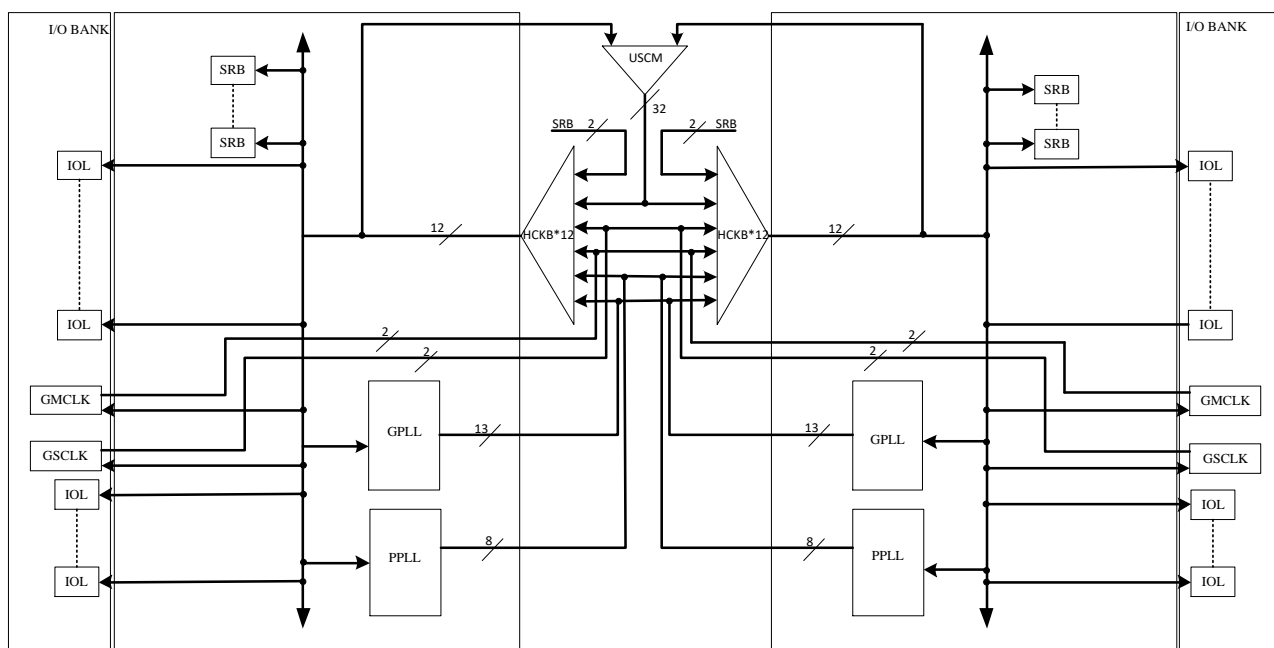


图 2-15 无 HSSTLP 区域水平时钟相邻区域 HCKB 输入时钟源和输出连接示意图

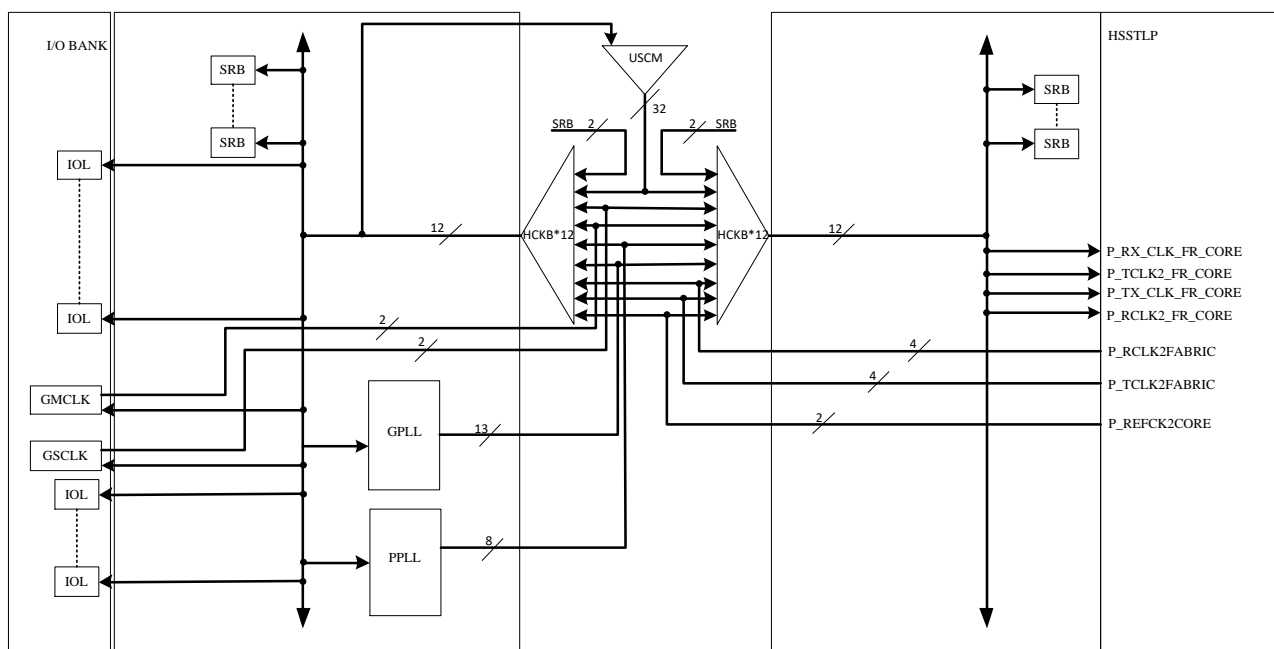


图 2-16 有 HSSTLP 区域水平相邻区域 HCKB 输入时钟源和输出连接示意图

HCKB 的输出时钟驱动 PLL、HSSTLP、SRB、IOL，如上图所示，HSSTLP 时钟端 P_REFCK2CORE 是差分输入参考时钟转单端后给 Fabric 的时钟。

2.3.2 水平时钟 GTP

一个水平时钟 GTP 占用一个 HCKB，每个区域同时最多支持使用 12 个。Logos2 系列产品拥有两种水平时钟 GTP，一种只有输入和输出的 GTP_CLKBUFX；另一种带有使能控制信号的 GTP_CLKBUFXCE，通过使能信号 CE 控制是否输出时钟信号，可以降低功耗。

如果设计模块只在一个区域或者相邻水平区域里，输入时钟可以不走全局时钟树，通过手动例化水平时钟 GTP 的方式，让输入时钟直接走 HCKB 驱动该模块逻辑，同时用户也可以根据自己设计需要将例化的原语约束到想要的位置来达到设计目的。

2.3.2.1 GTP_CLKBUFX

HCKB 的使用由用户例化 GTP_CLKBUFX 插入,该 CLKBUFX 无参数。以 Verilog 为例：

```
GTP_CLKBUFX  GTP_CLKBUFX_INST
(
    .CLKIN      (CLKIN      ),
    .CLKOUT     (CLKOUT     )
);
```

GTP_CLKBUFX 输入输出时序说明：输出时钟CLKOUT恒等于输入时钟CLKIN。

2.3.2.2 GTP_CLKBUFEXCE

GTP_CLKBUFEXCE 有使能控制信号 CE，控制是否输出时钟信号，通过改变参数可以设置使能模式（同步使能或者异步使能）、使能有效电平（高电平有效或者低电平有效）、以及触发模式（上升沿触发或者下降沿触发）。需要注意的是异步使能情况下，使能信号 CE 由高变成低电平时输出时钟会立刻拉低或者拉高，切换时可能出现毛刺，同步使能情况每次切换需要等待 2 个时钟沿才会切换，由触发模式决定上升沿或者下降沿。

GTP_CLKBUFEXCE 原语例化：

```
GTP_CLKBUFEXCE #(
    .CE_TYPE      ("SYNC"      ),
    .CE_INV       ("FALSE"     ),
    .TRIGGER_MODE ("POSEDGE"   )
) GTP_CLKBUFEXCE_INST (
    .CLKOUT       (CLKOUT      ),
    .CE           (CE          ),
    .CLKIN        (CLKIN       )
);
```

表 2-11 GTP_CLKBUFEXCE 端口描述

CLKIN0	输入	输入时钟
CLKIN	输入	输入时钟信号
CE	输入	时钟使能端
CLKOUT	输出	输出时钟信号

表 2-12 GTP_CLKBUFEXCE 参数描述

参数名称	参数类型	有效值	功能描述
CE_TYPE	<string>	“SYNC” “ASYN”	CE_TYPE= “SYNC”，支持同步使能。 CE_TYPE= “ASYN”，支持异步使能。
TRIGGER_MODE	<string>	“POSEDGE” “NEGEDGE”	TRIGGER_MODE = “POSEDGE”，支持 时钟输出初始值等于 1'b1，上升沿触发。 TRIGGER_MODE =“NEGEDGE”，支持 时钟输出初始值等于 1'b0，下降沿触发。
CE_INV	<string>	“TRUE” “FALSE”	CE_INV= “FALSE”，CE 高电平有效。 CE_INV=“TRUE”，CE 低电平有效。

同步使能模式：

当配置为上升沿触发同步使能时，输出时钟由输入时钟的上升沿触发，当使能无效时输出时钟保持高电平；当配置为下降沿触发同步使能时，输出时钟由输入时钟的下降沿触发，

当使能无效时输出时钟保持低电平。端口时序如下图所示：

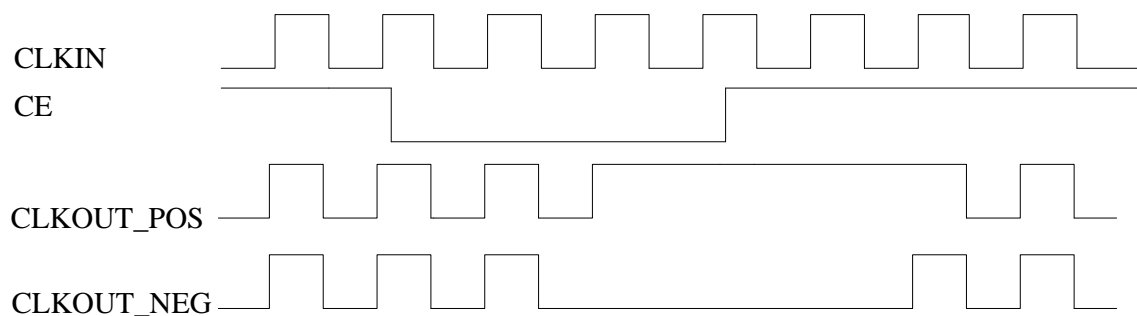


图 2-17 GTP_CLKBUFCE 端口时序图一

CLKOUT_POS代表的是TRIGGER_MODE = “POSEDGE”时的输出时钟，CLKOUT_NEG代表的是TRIGGER_MODE = “NEGEDGE”时的输出时钟，实线箭头所指是时钟切换时刻，CE默认电平是低电平。

异步使能模式：

当配置为异步使能时，使能信号CE切换时输出时钟立刻切换，这种情况下可能出现毛刺，端口时序如下图所示：

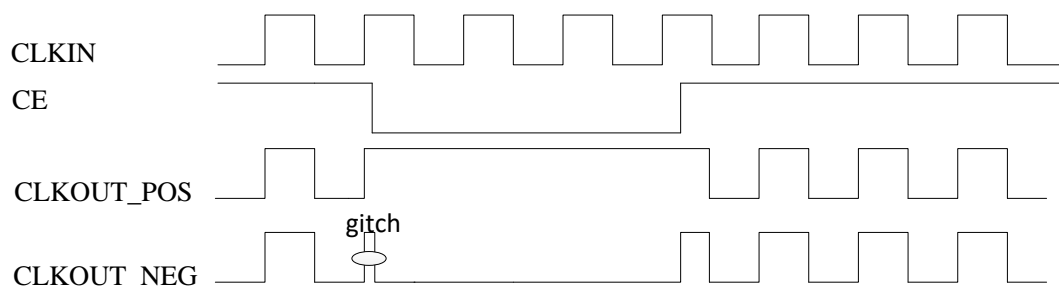


图 2-18 GTP_CLKBUFCE 端口时序图二

CLKOUT_POS代表的是TRIGGER_MODE = “POSEDGE”时的输出时钟，CLKOUT_NEG代表的是TRIGGER_MODE = “NEGEDGE”时的输出时钟，实线箭头所指是时钟切换时刻，CE默认电平是低电平。

2.4 区域时钟资源

2.4.1 区域时钟网络

REGIONAL CLK 基于 Region 分布,每个 Region 有 4 个 REGIONAL CLK 网络。区域时钟主要为单个 REGION 的逻辑单元提供同步时钟，由 RCKB 驱动。

下图是单个区域的RCKB作用域示意图,由于HSSTLP所在区域没有RCKB和MRCKB,

所以除 HSSTLP 所在区域外，其它区域 RCKB 输入输出连接关系均与该图一致。

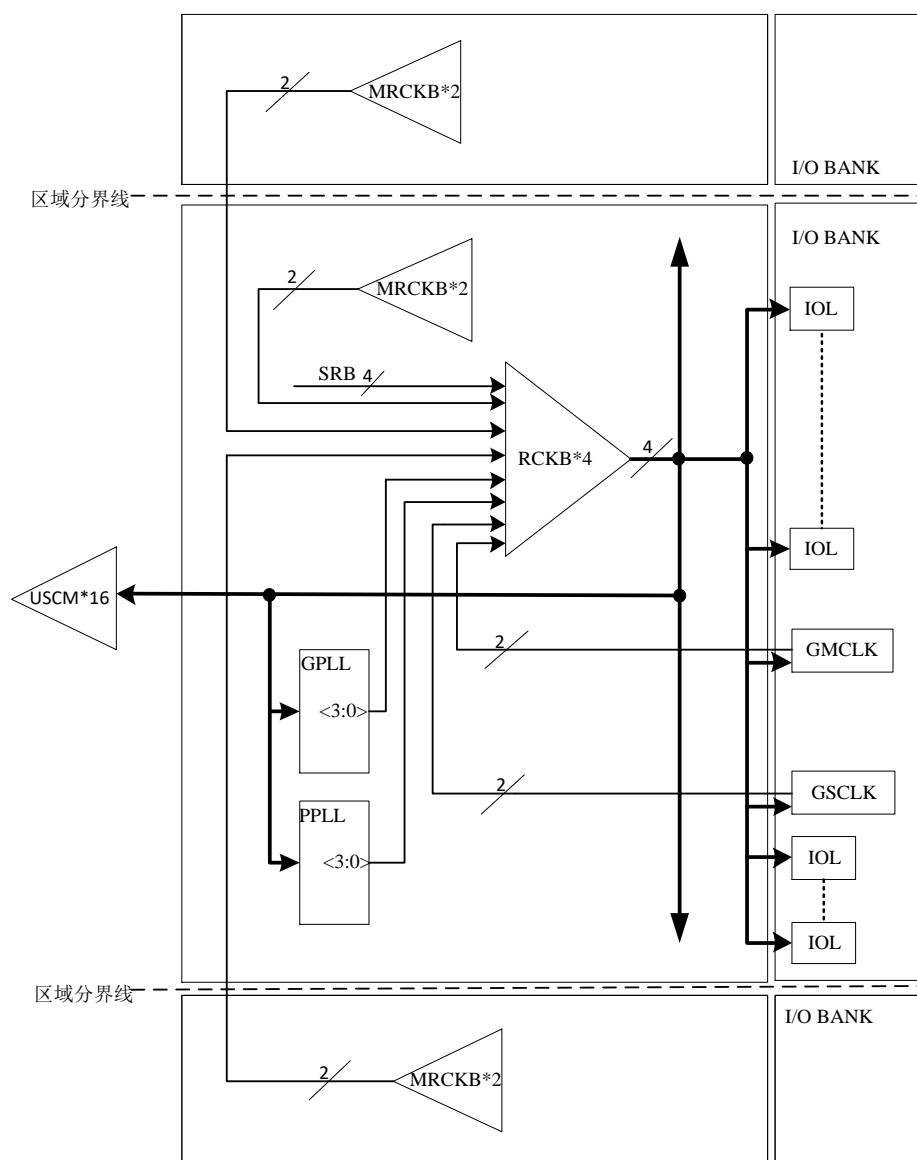


图 2-19 区域时钟资源与作用域示意图

2.4.2 区域时钟 GTP

2.4.2.1 GTP_CLKBUFR

RCKB 的使用由用户例化 GTP_CLKBUFR 插入,该 CLKBUFR 无参数。以 Verilog 为例:

```
GTP_CLKBUFR  GTP_CLKBUFR_INST
(
  .CLKIN      (CLKIN      ),
  .CLKOUT     (CLKOUT     )
);
```

GTP_CLKBUFR 端口时序如下图所示:

2.4.2.2 GTP_IOCLKDIV_E2

GTP_IOCLKDIV_E2 有 3 个输入端 CLKIN、RST_N、CE，1 个输出端 CLKDIVOUT 和 1 个参数 DIV_FACTOR。通过配置参数 DIV_FACTOR，可实现 1 分频到 8 分频和直通模式（BYPASS）。

- CE 为异步使能控制，当 CE = 0 时，CLKDIVOUT 保持当前电平；当 CE = 1 时，CLKDIVOUT 输出有效分频时钟
- RST_N 是复位信号，支持异步复位，同步释放，并非必须使用。当 RST_N = 0 时，对输出时钟清零
- BYPASS 模式下输出时钟不受 CE 或 RST_N 的控制

GTP_IOCLKDIV_E2 原语例化：

```
GTP_IOCLKDIV_E2 #(
.DIV_FACTOR    ("BYPASS"    )
) IOCLKDIV_E2_INST (
.CLKDIVOUT     (CLKDIVOUT   ),
.CE            (CE          ),
.CLKIN         (CLKIN       ),
.RST_N         (RST_N       )
);
```

表 2-13 GTP_IOCLKDIV_E2 端口描述

CLKIN0	输入	输入时钟
CLKIN	输入	输入时钟
RST_N	输入	复位信号，低电平有效
CE	输入	使能控制： CE = 1，输出分频后或不分频的时钟信号 CE = 0，输出时钟保持当前电平
CLKDIVOUT	输出	输出时钟

表 2-14 GTP_IOCLKDIV_E2 参数描述

参数名称	参数类型	有效值	功能描述
DIV_FACTOR	<string>	“BYPASS” “1” “2” “3” “4” “5” “6” “7” “8”	模式选择，包括直通模式和 1~8 分频模式

以输出时钟 1 分频和 2 分频为例，端口时序如下图所示：

当 RST_N 为低电平的时候，CLKOUT 输出为 0，当 RST_N 并且 CE 为高电平时，CLKOUT 输出有效时钟，当 CE 为低电平时，CLKOUT 立刻拉低，输出为 0。

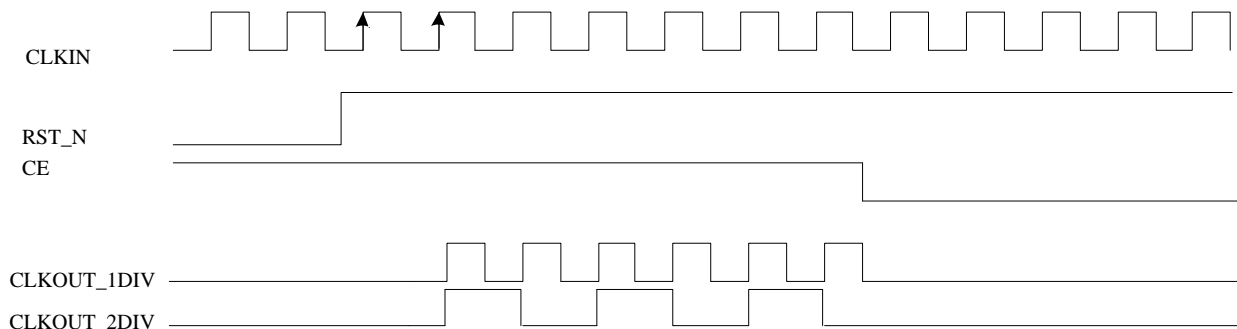


图 2-20 GTP_IOCLKDIV_E2 端口时序图

2.5 IO 时钟的资源

2.5.1 IO 时钟网络

IO CLK 为 Logos2 系列产品的 IO 提供了高速时钟，常用于高速接口逻辑，由 IOCKB 驱动。和 GCLK/RCLK 相比，IO CLK 具有频率高，延迟时间短，skew 小等特点。

PG2L100H 在每个 clock region 有四个 IOCKB 驱动 IO CLK 时钟线，如下图所示。

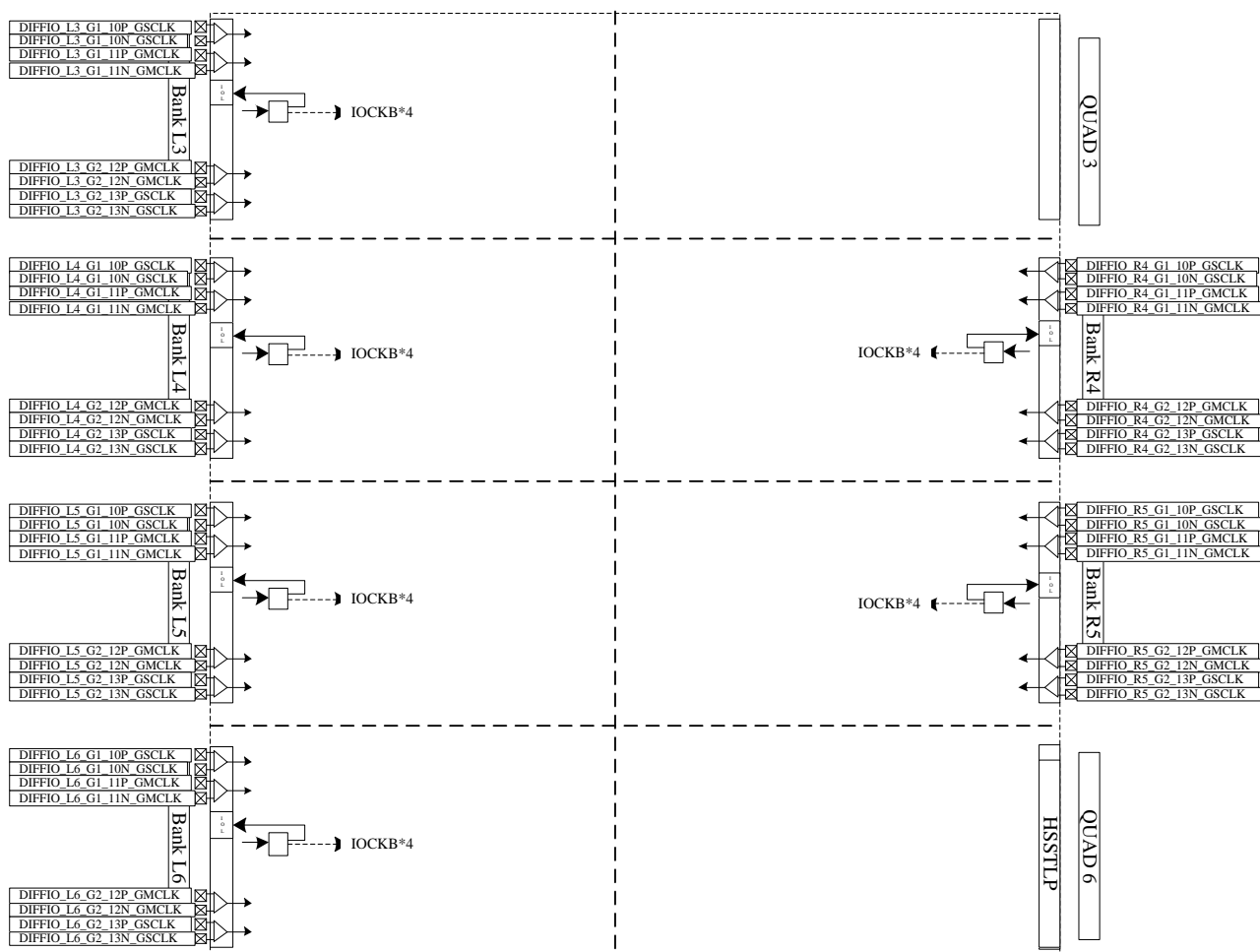


图 2-21 IO CLK 时钟资源示意图

如下图所示，PPLL 的输出时钟 CLKOUT0~3 经过 IOCKB 为 BANK 内的 IOL 提供高质量的高速时钟。

下图是单个区域 IOCKB 示意图，图示明确了 IOCKB 的输入时钟来源和输出驱动范围，由于 HSSTLP 所在区域没有 IOCKB 和 MRCKB，所以除 HSSTLP 所在区域外，其它区域 IOCKB 输入输出连接关系均与该图一致。

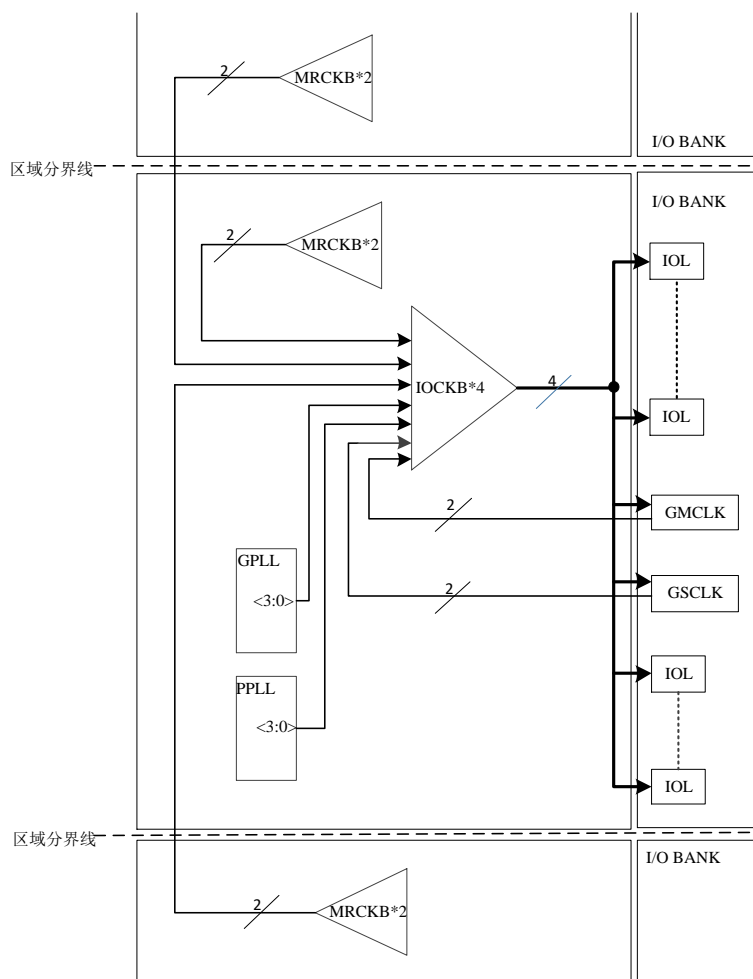


图 2-22 时钟输入时钟源连接示意图

2.5.2 IO 时钟 GTP

IOCKB 的使用由用户例化 GTP_IOCLKBUF 插入。以 Verilog 为例：

```
GTP_IOCLKBUF
```

```
#(
```

```
.GATE_EN ("FALSE" ) // Logos2 系列只支持参数“FALSE”
```

```
) GTP_IOCLKBUF_INST
```

```
(
```

```
.CLKIN (CLKIN ),
```

```

.DI          (DI          ),
.CLKOUT      (CLKOUT      )
);
    
```

表 2-15 GTP_IOCLKBUF 端口描述

端口信号	输入/输出	说明
CLKIN	输入	输入时钟
DI	输入	时钟使能，Logos2 不支持
CLKOUT	输出	输出时钟

表 2-16 GTP_IOCLKBUF 参数描述

参数名称	参数类型	有效值	功能描述
GATE_EN	<string>	<“FALSE”, “TRUE”>	Logos2 只能配置为“FALSE”

Logos2 系列器件的 GTP_IOCLKBUF 只支持 buffer 的功能，参数配置 GATE_EN 只能配置为“FALSE”，不支持 DI 控制时钟信号，即 DI 输入 1 或者 0，CLKOUT 都输出有效时钟信号。

2.6 跨区域时钟资源

2.6.1 跨区域时钟网络

MRCKB 主要提供时钟的跨时钟区域连接。如下图所示，MRCKB 的输入源有 GMCLK 和 SRB。

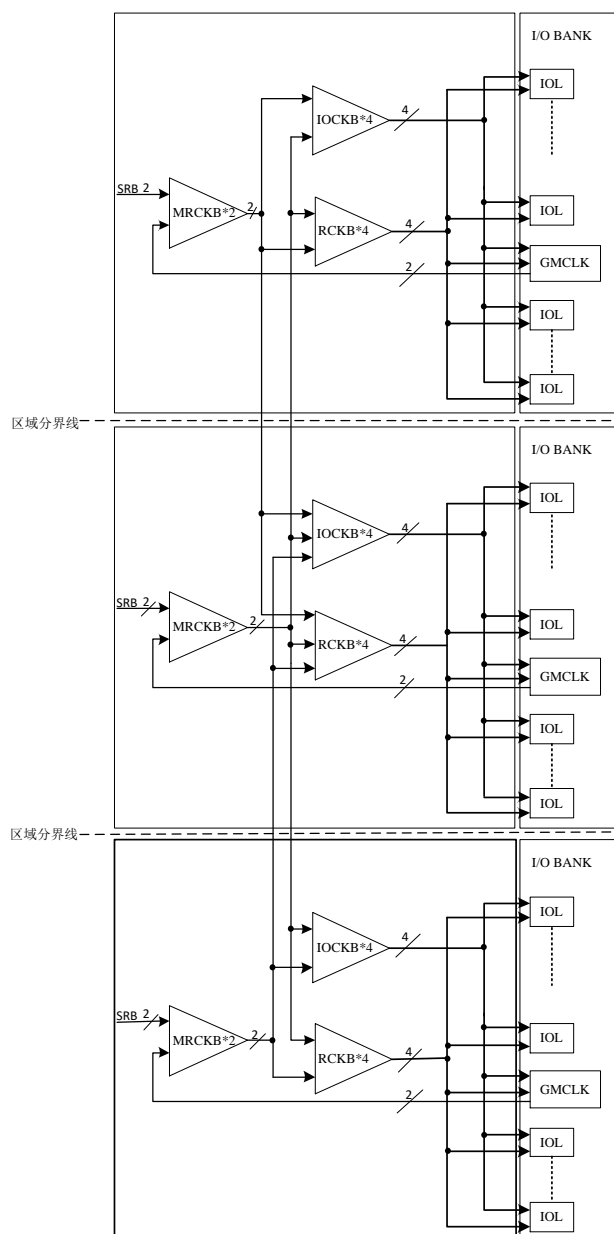


图 2-23 MRCKB 输入时钟源和输出连接关系示意图

跨区域时钟 buffer 输入时钟源和输出连接示意图如上图所示，图示明确了 MRCKB 的输入时钟来源、输出驱动资源和可到达区域，由于 HSSTLP 所在区域没有 MRCKB、IOCKB 和 RCKB，所以除 HSSTLP 所在区域外，其它区域 MRCKB 连接关系均与该图一致。当外部时钟单端输入时，只能使用 GMCLK_P 端，才能保证时钟信号走专用时钟路径到达 MRCKB。

2.6.2 跨区域时钟 GTP

2.6.2.1 GTP_CLKBUFM

MRCKB 的支持由用户例化 GTP_CLKBUFM 或 GTP_CLKBUFMCE 插入,以 Verilog 为例，当用户例化 GTP_CLKBUFM 插入 MRCKB 时，如下：

GTP_CLKBUFM GTP_CLKBUFM_INST

```
(
    .CLKIN      (CLKIN      ),
    .CLKOUT     (CLKOUT     ));
```

GTP_CLKBUFM 输入输出时序说明：输出时钟CLKOUT恒等于输入时钟CLKIN。

2.6.2.2 GTP_CLKBUFMCE

当用户例化带有使能端的GTP_CLKBUFMCE时插入MRCKB时，以verilog为例。

```
GTP_CLKBUFMCE #(
    .CE_TYPE      ("SYNC"      ),
    .CE_INV       ("FALSE"     ),
    .TRIGGER_MODE ("POSEDGE"   )
) CLKBUFMCE_INST (
    .CLKOUT       (CLKOUT      ),
    .CE           (CE          ),
    .CLKIN        (CLKIN       ));
```

表 2-17 GTP_CLKBUFMCE 端口描述

端口名称	输入输出	功能描述
CLKIN	输入	输入时钟信号
CE	输入	时钟使能端
CLKOUT	输出	输出时钟信号

表 2-18 GTP_CLKBUFMCE 参数描述

参数名称	有效值	功能描述
CE_TYPE	“SYNC” “ASYN”	CE_TYPE= “SYNC”，支持同步使能 CE_TYPE= “ASYN”，支持异步使能
TRIGGER_MODE	“POSEDGE” “NEGEDGE”	TRIGGER_MODE = “POSEDGE”，支持时钟输出初始值等于1'b1，上升沿触发 TRIGGER_MODE =“NEGEDGE”，支持时钟输出初始值等于1'b0，下降沿触发
CE_INV	“TRUE” “FALSE”	CE_INV= “FALSE”，默认 CE 高电平有效 CE_INV=“TRUE”，默认 CE 低电平有效

表 2-19 GTP_CLKBUFMCE 模式描述

CE_TYPE	TRIGGER_MODE	CE_INV	功能
“SYNC”	“POSEDGE”	“FALSE”	上升沿同步使能模式
“SYNC”	“NEGEDGE”	“FALSE”	下降沿同步使能模式

CE_TYPE	TRIGGER_MODE	CE_INV	功能
“ASYNC”	“NEGEDGE”	“FALSE”	异步使能模式 0
“ASYNC”	“POSEDGE”	“FALSE”	异步使能模式 1

GTP_CLKBUFMCE不同工作模式输入输出时序说明:

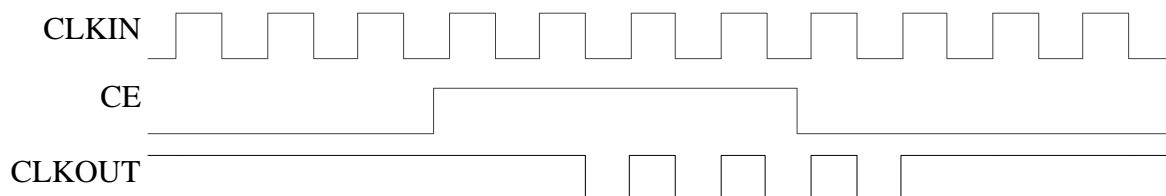


图 2-24 GTP_CLKBUFMCE 上升沿同步使能输入输出时序图

当配置为上升沿触发同步使能时，输出时钟由输入时钟的上升沿触发，当使能无效时输出时钟保持高电平。

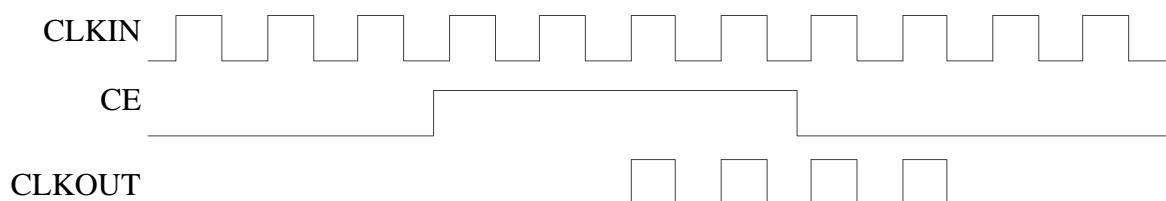


图 2-25 GTP_CLKBUFMCE 下降沿同步使能输入输出时序图

当配置为下降沿触发同步使能时，输出时钟由输入时钟的下降沿触发，当使能无效时输出时钟保持低电平。

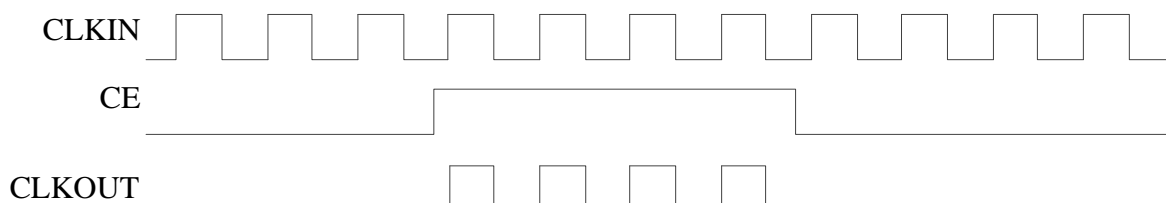


图 2-26 GTP_CLKBUFMCE 异步使能模式 0 输入输出时序图

当配置为异步使能模式0，使能失效时输出时钟输出低电平，功能时序图如上图所示。

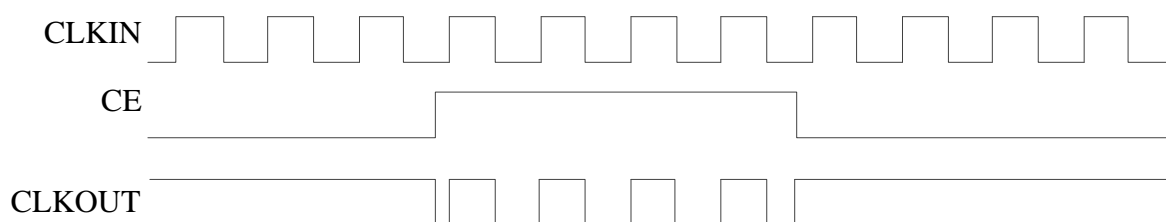


图 2-27 GTP_CLKBUFMCE 异步使能模式 1 输入输出时序图

当配置为异步使能模式1，使能失效时输出时钟输出高电平，功能时序图如上图所示。

2.7 PLL 时钟连接

GPLL 和 PPLL 的输入源有：

- 相同左侧或右侧的 GMCLK 和 GSCLK
- USCM
- 相同区域的 RCKB
- 相同区域的 HCKB
- GPLL/PPLL CLKOUT0~3/N
- HSSTLP 时钟，详见 1.2 小节时钟连接概括

如下图所示 GPLL 和 PPLL 时钟源连接，除有 HSSTLP 的水平相邻区域外，其它 Bank 时钟源连接相同。

由于 HSSTLP 区域没有 GPLL 和 PPLL，HSSTLP 输出时钟想作为其输入时钟源必须通过 USCM 或者 HCKB 到达 GPLL/PPLL，可插入 GTP_CLKBUFG 或 GTP_CLKBUFX。

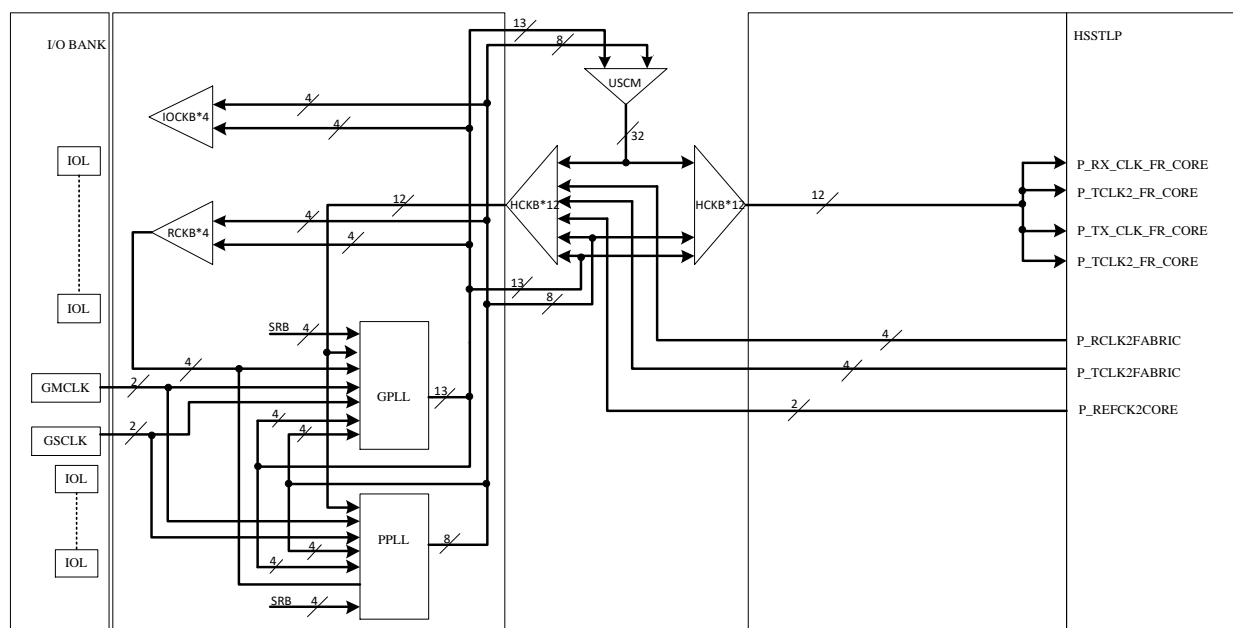


图 2-28 PLL 输入时钟源和输出时钟连接示意图

通过 PLL 的时钟主干路或时钟树（RCKB、HCKB），PPLL、GPLL 的 CLKOUT0~3 可作为其他 PLL 里的 PPLL、GPLL 的输入时钟，进而完成 PLL 的级联。

注：

1. 时钟从时钟管脚 GMCLK 和 GSCLK 进来也能到达左右半区的 GPLL/PPLL，但是要经过 USCM 或者相邻水平区域的 HCKB 才能过去
2. GPLL 的输出时钟 CLKOUT0N~3N 和 PPLL 的输出时钟 CLKOUT0N 单独使用时只能驱动 USCM 和 HCKB，PPLL 的输出时钟 CLKOUT1N~3N 不能作为用户时钟单独使用

2.8 GPLL 使用说明

2.8.1 GPLL 概述

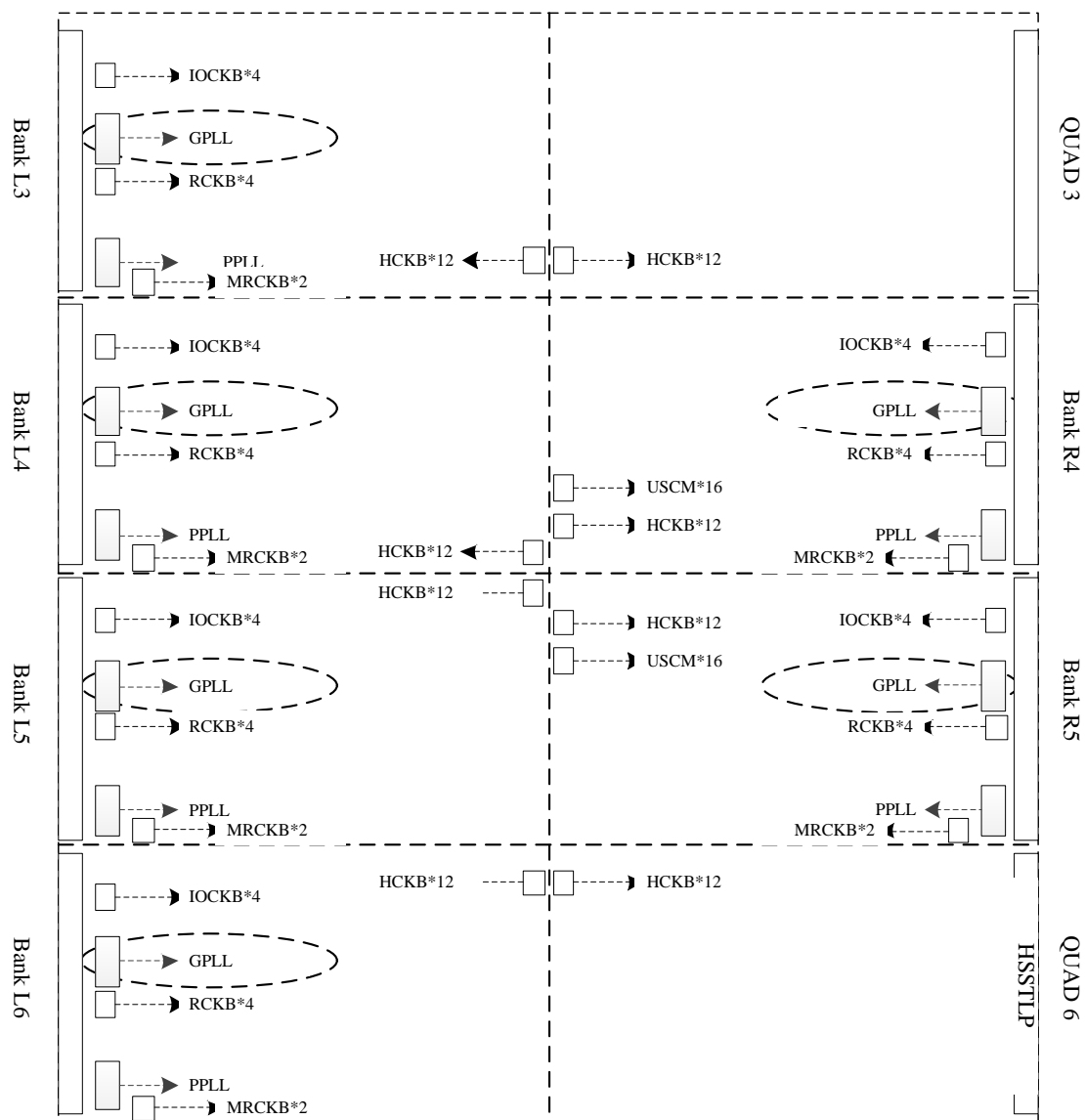


图 2-29 GPLL 资源示意图 (Top View)

PGL2G100H 内部有 6 个 GPLL，每个时钟区域都有一个 GPLL，具体如上图所示。GPLL 主要有以下特性：时钟频率综合、降低时钟 SKEW、相位调整、扩频输出、时钟级联、输出时钟门控、APB 动态重配等。

GTP_GPLL 系统框图如下图所示：

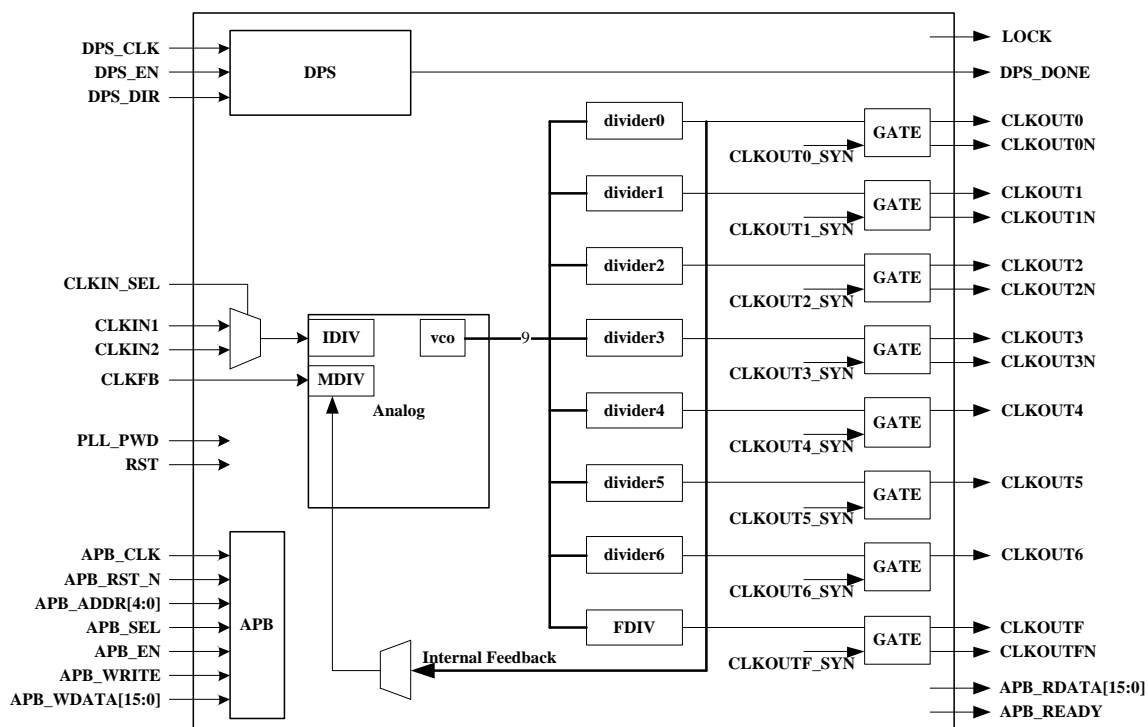


图 2-30 GPLL 系统框图

在使用中通过调用 GTP_GPLL 的实例配置使用，对于不用的输出端口可以配置输出关断来降低功耗，下面介绍 GTP_GPLL 的应用。

2.8.2 GTP_GPLL 顶层框图

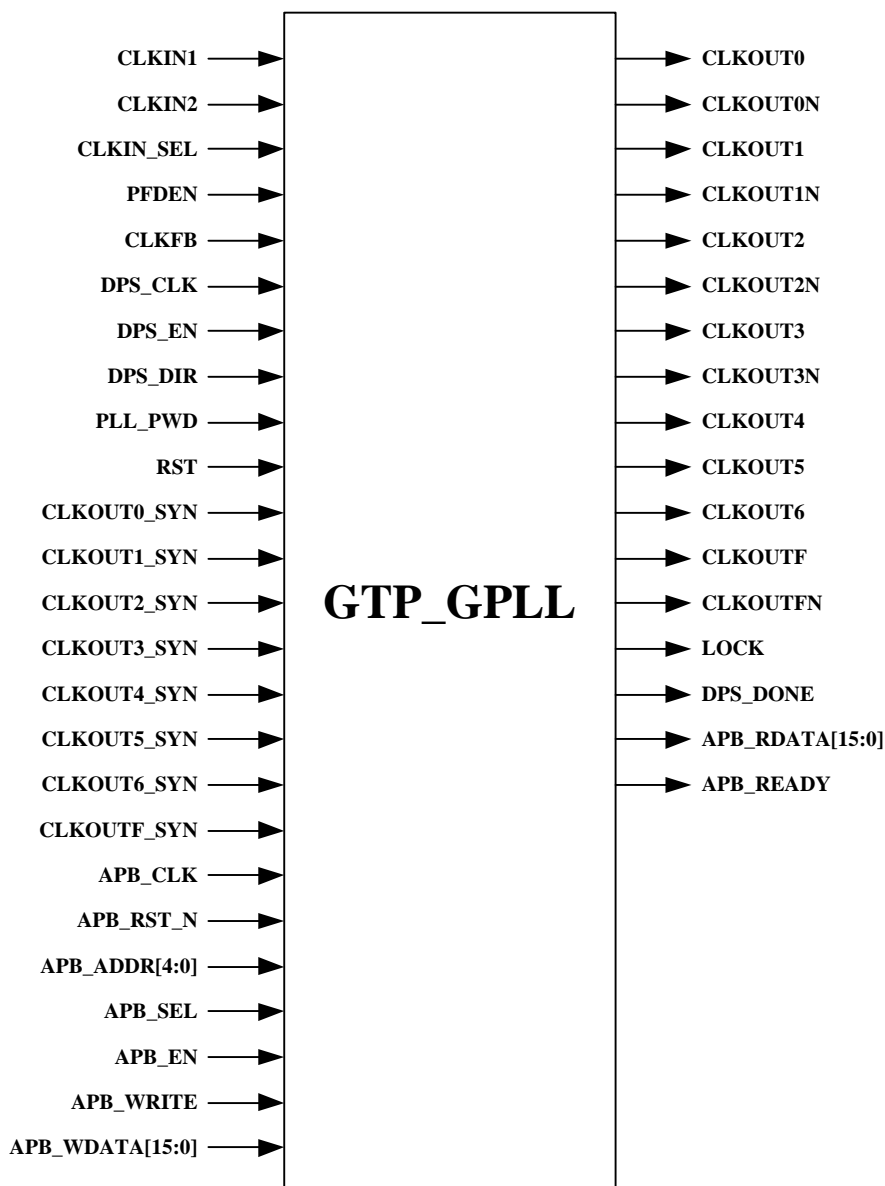


图 2-31 GTP_GPLL 框图

2.8.3 GTP_GPLL 端口列表

表 2-20 GTP_GPLL 端口描述

Port	I/O	功能描述
CLKOUT0	O	PLL 第 0 路正相输出时钟;
CLKOUT0N	O	PLL 第 0 路反相输出时钟;
CLKOUT1	O	PLL 第 1 路正相输出时钟;
CLKOUT1N	O	PLL 第 1 路反相输出时钟;
CLKOUT2	O	PLL 第 2 路正相输出时钟;
CLKOUT2N	O	PLL 第 2 路反相输出时钟;
CLKOUT3	O	PLL 第 3 路正相输出时钟;

Port	I/O	功能描述
CLKOUT3N	O	PLL 第 3 路反相输出时钟；
CLKOUT4	O	PLL 第 4 路输出时钟；
CLKOUT5	O	PLL 第 5 路输出时钟；
CLKOUT6	O	PLL 第 6 路输出时钟；
CLKOUTF	O	PLL 反馈正相输出时钟；
CLKOUTFN	O	PLL 反馈反相输出时钟；
LOCK	O	PLL 频率锁定指示信号，为异步信号； 信号拉高时，指示 PLL 反馈时钟信号已锁定到输入时钟信号；
DPS_DONE	O	动态插值相移调整指示信号；
APB_RDATA[15:0]	O	PLL APB 接口数据总线数据输出；
APB_READY	O	PLL APB 接口数据总线握手信号；表示一个正常的总线周期结束；
CLKIN1	I	PLL 参考输入时钟 1；
CLKIN2	I	PLL 参考输入时钟 2；
CLKFB	I	PLL 反馈时钟；
CLKIN_SEL	I	输入时钟选择信号；
DPS_CLK	I	动态插值相移调整时钟；
DPS_EN	I	动态插值相移调整使能；高电平有效；
DPS_DIR	I	选择动态插值相移调整方向；1'b0：滞后，1'b1：超前；
PLL_PWD	I	PLL Power Down；高电平有效；
RST	I	PLL 复位信号，高电平有效；
CLKOUT0_SYN	I	CLKOUT0/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT1_SYN	I	CLKOUT1/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT2_SYN	I	CLKOUT2/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT3_SYN	I	CLKOUT3/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT4_SYN	I	CLKOUT4 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT5_SYN	I	CLKOUT5 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT6_SYN	I	CLKOUT6 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUTF_SYN	I	CLKOUTF/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
APB_CLK	I	PLL APB 接口数据总线时钟；
APB_RST_N	I	PLL APB 接口数据总线异步复位信号，低电平有效；
APB_ADDR	I	PLL APB 接口数据总线地址；
APB_SEL	I	PLL APB 接口数据总线选择信号，选择从设备；
APB_EN	I	PLL APB 接口数据总线使能信号；
APB_WRITE	I	PLL APB 接口数据总线写使能信号；1'b0：读操作，1'b1：写操作；
APB_WDATA[15:0]	I	PLL APB 接口数据总线数据输入；

2.8.4 GTP_GPLL 参数列表

表 2-21 GTP_GPLL 参数描述

参数名称	参数类型	有效值	功能描述
CLKIN_FREQ	real	10~800	输入时钟频率配置, MHz;
LOCK_MODE	binary	1'b0,1'b1	PLL 频率检测模式配置; 1'b0: 实时监测 PLL 工作状态; 1'b1: PLL 锁定后 LOCK 一直保持, 除非复位或 Power Down;
STATIC_RATIOI	integer	1-80	输入 divider ratio 静态配置;
STATIC_RATIOM	integer	1-128	反馈 M divider ratio 静态配置;
STATIC_RATIO0	real	1-128 or 2.000~128.000, step 0.125	输出 divider0 ratio 静态配置; 整数分频模式, 支持 1~128; 分数分频模式, 支持 2.000~128.000;
STATIC_RATIO1	integer	1-128	输出 divider1 ratio 静态配置;
STATIC_RATIO2	integer	1-128	输出 divider2 ratio 静态配置;
STATIC_RATIO3	integer	1-128	输出 divider3 ratio 静态配置;
STATIC_RATIO4	integer	1-128	输出 divider4 ratio 静态配置;
STATIC_RATIO5	integer	1-128	输出 divider5 ratio 静态配置;
STATIC_RATIO6	integer	1-128	输出 divider6 ratio 静态配置;
STATIC_RATIOF	real	1-128 or 2.000~128.000 step 0.125	反馈 F divider ratio 静态配置; 整数分频模式, 支持 1~128; 分数分频模式, 支持 2.000~128.000;
STATIC_DUTY0	integer	2~255	输出 divider0 duty 静态配置;
STATIC_DUTY1	integer	2~255	输出 divider1 duty 静态配置;
STATIC_DUTY2	integer	2~255	输出 divider2 duty 静态配置;
STATIC_DUTY3	integer	2~255	输出 divider3 duty 静态配置;
STATIC_DUTY4	integer	2~255	输出 divider4 duty 静态配置;
STATIC_DUTY5	integer	2~255	输出 divider5 duty 静态配置;
STATIC_DUTY6	integer	2~255	输出 divider6 duty 静态配置;
STATIC_DUTYF	integer	2~255	反馈 F divider duty 静态配置;
STATIC_PHASE	integer	0-63	插值相移 phase 静态配置;
STATIC_PHASE0	integer	0-7	输出 divider0 fine phase 静态配置;
STATIC_PHASE1	integer	0-7	输出 divider1 fine phase 静态配置;
STATIC_PHASE2	integer	0-7	输出 divider2 fine phase 静态配置;
STATIC_PHASE3	integer	0-7	输出 divider3 fine phase 静态配置;
STATIC_PHASE4	integer	0-7	输出 divider4 fine phase 静态配置;
STATIC_PHASE5	integer	0-7	输出 divider5 fine phase 静态配置;
STATIC_PHASE6	integer	0-7	输出 divider6 fine phase 静态配置;
STATIC_PHASEF	integer	0-7	反馈 F divider fine phase 静态配置;

参数名称	参数类型	有效值	功能描述
STATIC_CPHASE0	integer	0-127	输出 divider0 coarse phase 静态配置;
STATIC_CPHASE1	integer	0-127	输出 divider1 coarse phase 静态配置;
STATIC_CPHASE2	integer	0-127	输出 divider2 coarse phase 静态配置;
STATIC_CPHASE3	integer	0-127	输出 divider3 coarse phase 静态配置;
STATIC_CPHASE4	integer	0-127	输出 divider4 coarse phase 静态配置;
STATIC_CPHASE5	integer	0-127	输出 divider5 coarse phase 静态配置;
STATIC_CPHASE6	integer	0-127	输出 divider6 coarse phase 静态配置;
STATIC_CPHASEF	integer	0-127	反馈 F divider coarse phase 静态配置;
CLK_DPS0_EN	string	"FALSE", "TRUE"	输出 divider0 插值相移使能;
CLK_DPS1_EN	string	"FALSE", "TRUE"	输出 divider1 插值相移使能;
CLK_DPS2_EN	string	"FALSE", "TRUE"	输出 divider2 插值相移使能;
CLK_DPS3_EN	string	"FALSE", "TRUE"	输出 divider3 插值相移使能;
CLK_DPS4_EN	string	"FALSE", "TRUE"	输出 divider4 插值相移使能;
CLK_DPS5_EN	string	"FALSE", "TRUE"	输出 divider5 插值相移使能;
CLK_DPS6_EN	string	"FALSE", "TRUE"	输出 divider6 插值相移使能;
CLK_DPSF_EN	string	"FALSE", "TRUE"	反馈 F divider 插值相移使能;
CLK_CAS5_EN	string	"FALSE", "TRUE"	输出 divider5 时钟级联使能;
CLKOUT0_SYN_EN	string	"FALSE", "TRUE"	CLKOUT0_SYN 信号使能配置;
CLKOUT1_SYN_EN	string	"FALSE", "TRUE"	CLKOUT1_SYN 信号使能配置;
CLKOUT2_SYN_EN	string	"FALSE", "TRUE"	CLKOUT2_SYN 信号使能配置;
CLKOUT3_SYN_EN	string	"FALSE", "TRUE"	CLKOUT3_SYN 信号使能配置;
CLKOUT4_SYN_EN	string	"FALSE", "TRUE"	CLKOUT4_SYN 信号使能配置;
CLKOUT5_SYN_EN	string	"FALSE", "TRUE"	CLKOUT5_SYN 信号使能配置;
CLKOUT6_SYN_EN	string	"FALSE", "TRUE"	CLKOUT6_SYN 信号使能配置;
CLKOUTF_SYN_EN	string	"FALSE", "TRUE"	CLKOUTF_SYN 信号使能配置;
SSC_MODE	string	"DOWN_LOW", "DOWN_HIGH", "CENTER_LOW", "CENTER_HIGH", "DISABLE"	SSC 模式配置;
SSC_FREQ	real	25~250	SSC 调制频率配置, KHz;
INTERNAL_FB	string	"CLKOUT0", "CLKOUT1", "CLKOUT2", "CLKOUT3", "CLKOUT4", "CLKOUT5", "CLKOUT6", "CLKOUTF", "DISABLE"	内部反馈路径选择, 选择这几个输出时钟其中一个作为内反馈时钟, 如 INTERNAL_FB = "CLKOUT0", 则输出时钟 CLKOUT0 作为内反馈时钟;
EXTERNAL_FB	string	"CLKOUT0", "CLKOUT1"	外部反馈路径选择, 选择这几个输出时钟其中一个作为外反馈时钟如

参数名称	参数类型	有效值	功能描述
		“CLKOUT2”, “CLKOUT3”, “CLKOUT4”, “CLKOUT5”, “CLKOUT6”, “CLKOUTF”, “DISABLE”	EXTERNAL_FB = " CLKOUT0", 则输出时钟 CLKOUT0 作为外反馈时钟;
BANDWIDTH	string	“LOW”, “HIGH” “OPTIMIZED”	带宽选择配置: BANDWIDTH=“LOW” 配置为“LOW”, BANDWIDTH=“HIGH” 配置为“HIGH” BANDWIDTH=“OPTIMIZED” 配置“OPTIMIZED”;

注:

1. 参数 CLK_DPS5_EN 和参数 CLK_CAS5_EN 不能同时配置为“TRUE”
2. 参数 INTERNAL_FB 和参数 EXTERNAL_FB 不能同时配置为“DISABLE”

2.8.5 GTP_GPLL APB 接口配置

用户逻辑通过 APB 接口对 GTP_GPLL 工作参数进行动态地改变。APB 接口读写寄存器时序如下图所示:

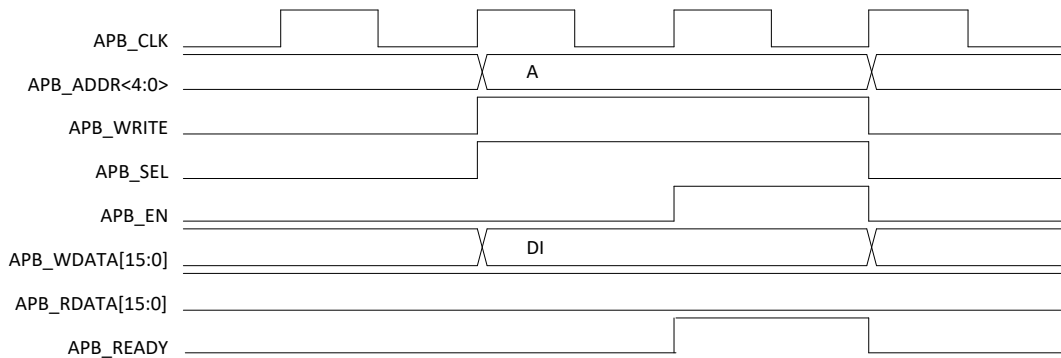


图 2-32 GTP_GPLL APB 写时序

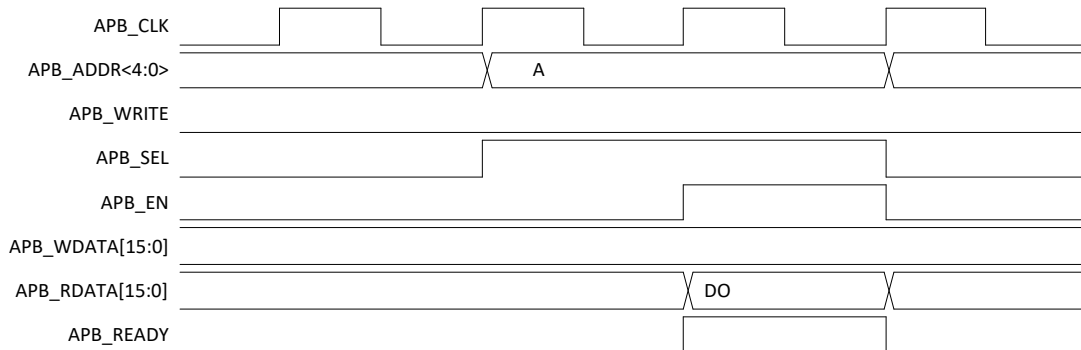


图 2-33 GTP_GPLL APB 读时序

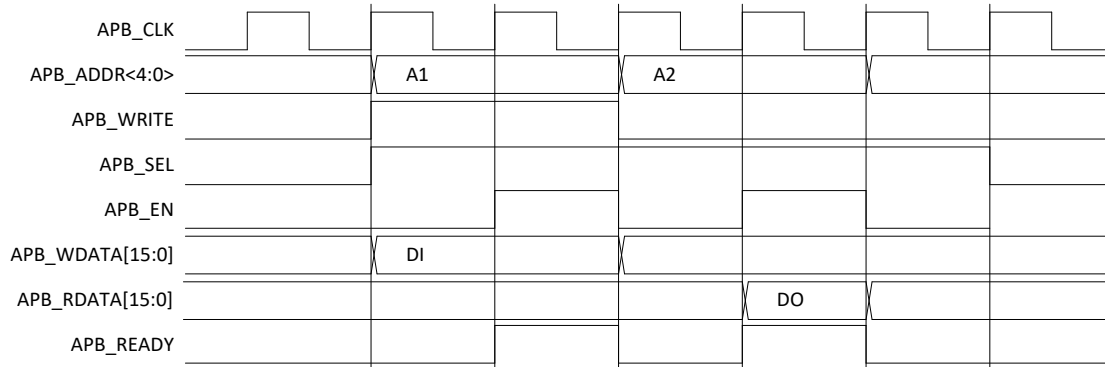


图 2-34 GTP_GPLL_APB 写后读时序

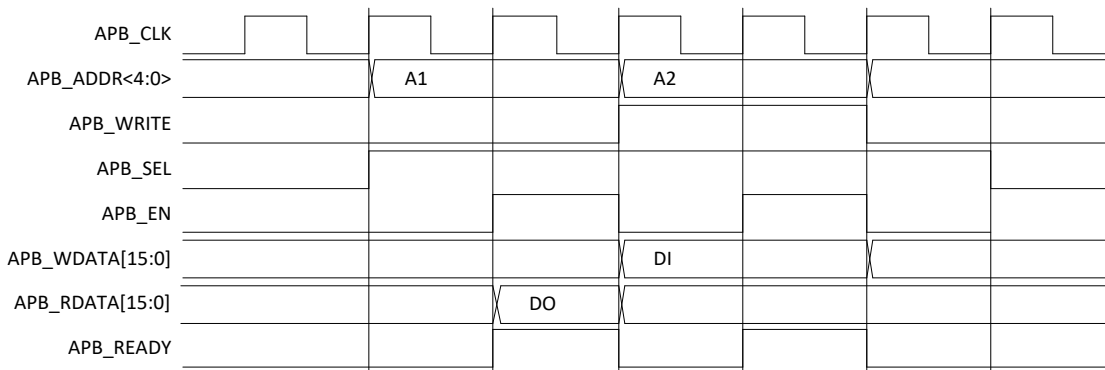


图 2-35 GTP_GPLL_APB 接口读后写时序

表 2-22 GTP_GPLL_APB 端口描述

序号	端口名称	I/O	描述
1	APB_CLK	I	时钟，上升沿采样；
2	APB_RST_N	I	异步复位，低有效；只复位 Data Bus（不复位寄存器）；
3	APB_ADDR[4:0]	I	地址总线；
4	APB_SEL	I	选择，表示从设备被选择；
5	APB_EN	I	使能，总线使能信号；
6	APB_WRITE	I	方向，0：读，1：写；
7	APB_WDATA[15:0]	I	数据总线输入；
8	APB_RDATA[15:0]	O	数据总线输出；
9	APB_READY	O	标志信号，高有效；

通过 APB 接口写内部寄存器的系统应用要求：

- APB 改写内部寄存器后，需要复位 PLL，保证其正常工作
- T_{RST} 最小脉冲宽度性能指标见 《DS04001_Logos2 系列 FPGA 器件数据手册》

表 2-23 GTP_GPLL APB 接口寄存器描述

APB_ADDR[4:0]	Bit	寄存器	描述
0	[15:8]	REG_ODIV0_RATIO[7:0]	ODIV0 Register1: 分频比设置;
	[7:0]	REG_ODIV0_DUTY[7:0]	ODIV0 Register1: 占空比设置;
1	[15:13]	REG_ODIV0_FRACDIV_RATIO[2:0]	ODIV0 Register2: 分数分频的分数设置, 范围;
	[12:10]	REG_ODIV0_FRACDIV_PSOFFSET[2:0]	ODIV0 Register2: 分数分频下降沿相位设置;
	[9:7]	REG_ODIV0_FPHASE[2:0]	ODIV0 Register2: 相位细调设置;
	[6:0]	REG_ODIV0_CPHASE[6:0]	ODIV0 Register2: 相位粗调设置;
2	[15:6]	RESERVED	ODIV0 Register3
	[5:4]	REG_ODIV0_ADJUST[1:0]	ODIV0 Register3: 分数分频时序调整设置;
	[3]	REG_ODIV0_FRACDIV_EN	ODIV0 Register3: 分数分频使能;
	[2:1]	REG_ODIV0_DPSORCAS_SEL[1:0]	ODIV0 Register3: 输入时钟选择;
	[0]	REG_ODIV0_MUXSEL_EN	ODIV0 Register3: 输入时钟 MUX 使能;
3	[15:8]	REG_ODIV1_RATIO[7:0]	ODIV1 Register1: 分频比设置;
	[7:0]	REG_ODIV1_DUTY[7:0]	ODIV1 Register1: 占空比设置;
4	[15:13]	RESERVED	
	[12:11]	REG_ODIV1_DPSORCAS_SEL[1:0]	ODIV1 Register2: 输入时钟选择;
	[10]	REG_ODIV1_MUXSEL_EN	ODIV1 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV1_FPHASE[2:0]	ODIV1 Register2: 相位细调设置;
	[6:0]	REG_ODIV1_CPHASE[6:0]	ODIV1 Register2: 相位粗调设置;
5	[15:8]	REG_ODIV2_RATIO[7:0]	ODIV2 Register1: 分频比设置;
	[7:0]	REG_ODIV2_DUTY[7:0]	ODIV2 Register1: 占空比设置;
6	[15:13]	RESERVED	
	[12:11]	REG_ODIV2_DPSORCAS_SEL[1:0]	ODIV2 Register2: 输入时钟选择;
	[10]	REG_ODIV2_MUXSEL_EN	ODIV2 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV2_FPHASE[2:0]	ODIV2 Register2: 相位细调设置;
	[6:0]	REG_ODIV2_CPHASE[6:0]	ODIV2 Register2: 相位粗调设置;
7	[15:8]	REG_ODIV3_RATIO[7:0]	ODIV3 Register1: 分频比设置;
	[7:0]	REG_ODIV3_DUTY[7:0]	ODIV3 Register1: 占空比设置;
8	[15:13]	RESERVED	
	[12:11]	REG_ODIV3_DPSORCAS_SEL[1:0]	ODIV3 Register2: 输入时钟选择;
	[10]	REG_ODIV3_MUXSEL_EN	ODIV3 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV3_FPHASE[2:0]	ODIV3 Register2: 相位细调设置;
	[6:0]	REG_ODIV3_CPHASE[6:0]	ODIV3 Register2: 相位粗调设置;
9	[15:8]	REG_ODIV4_RATIO[7:0]	ODIV4 Register1: 分频比设置;
	[7:0]	REG_ODIV4_DUTY[7:0]	ODIV4 Register1: 占空比设置;

APB_ADDR[4:0]	Bit	寄存器	描述
A	[15:13]	RESERVED	
	[12:11]	REG_ODIV4_DPSORCAS_SEL[1:0]	ODIV4 Register2: 输入时钟选择;
	[10]	REG_ODIV4_MUXSEL_EN	ODIV4 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV4_FPHASE[2:0]	ODIV4 Register2: 相位细调设置;
	[6:0]	REG_ODIV4_CPHASE[6:0]	ODIV4 Register2: 相位粗调设置;
B	[15:8]	REG_ODIV5_RATIO[7:0]	ODIV5 Register1: 分频比设置;
	[7:0]	REG_ODIV5_DUTY[7:0]	ODIV5 Register1: 占空比设置;
C	[15:13]	RESERVED	
	[12:11]	REG_ODIV5_DPSORCAS_SEL[1:0]	ODIV5 Register2: 输入时钟选择;
	[10]	REG_ODIV5_MUXSEL_EN	ODIV5 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV5_FPHASE[2:0]	ODIV5 Register2: 相位细调设置;
	[6:0]	REG_ODIV5_CPHASE[6:0]	ODIV5 Register2: 相位粗调设置;
D	[15:8]	REG_ODIV6_RATIO[7:0]	ODIV6 Register1: 分频比设置;
	[7:0]	REG_ODIV6_DUTY[7:0]	ODIV6 Register1: 占空比设置;
E	[15:13]	RESERVED	
	[12:11]	REG_ODIV6_DPSORCAS_SEL[1:0]	ODIV6 Register2: 输入时钟选择;
	[10]	REG_ODIV6_MUXSEL_EN	ODIV6 Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIV6_FPHASE[2:0]	ODIV6 Register2: 相位细调设置;
	[6:0]	REG_ODIV6_CPHASE[6:0]	ODIV6 Register2: 相位粗调设置;
F	[15:8]	REG_FDIV_RATIO[7:0]	FDIV Register1: 分频比设置;
	[7:0]	REG_FDIV_DUTY[7:0]	FDIV Register1: 占空比设置;
10	[15:13]	REG_FDIV_FRACDIV_RATIO[2:0]	FDIV Register2: 分数分频的分数设置;
	[12:10]	REG_FDIV_FRACDIV_PSOFFSET[2:0]	FDIV Register2: 分数分频下降沿相位设置;
	[9:7]	REG_FDIV_FPHASE[2:0]	FDIV Register2: 相位细调设置;
	[6:0]	REG_FDIV_CPHASE[6:0]	FDIV Register2: 相位粗调设置;
11	[15:6]	RESERVED	FDIV Register3
	[5:4]	REG_FDIV_ADJUST[1:0]	FDIV Register3: 分数分频时序调整设置;
	[3]	REG_FDIV_FRACDIV_EN	FDIV Register3: 分数分频使能;
	[2:1]	REG_FDIV_DPSORCAS_SEL[1:0]	FDIV Register3: 输入时钟选择;
	[0]	REG_FDIV_MUXSEL_EN	FDIV Register3: 输入时钟 MUX 使能;
12	[15:8]	RESERVED	
	[7:0]	REG_IDIV_RATIO[7:0]	IDIV Register: 分频比设置;
13	[15:8]	RESERVED	
	[7:0]	REG_MDIV_RATIO[7:0]	MDIV Register: 分频比设置;
14	[15:14]	RESERVED	BANDWIDTH Register

APB_ADDR[4:0]	Bit	寄存器	描述
	[13:12]	REG_VCTRL_INIT[1:0]	
	[11:10]	REG_CP_SELFBIAS_SEL[1:0]	
	[9:8]	REG_ICP_BASE_SEL[1:0]	
	[7:4]	REG_CP_CUR_SEL<3:0>	
	[3:1]	REG_LPF_R[2:0]	
	[0]	REG_LPF_C	
15	[15:6]	RESERVED	LOCK Register
	[5]	REG_LOCK_FILTER_PD	
	[4:0]	REG_FREQ_LOCKDET_SET<4:0>	

2.8.6 GTP_GPLL 功能说明

2.8.6.1 时钟输入

CLKIN_SEL 为 0 时，选择 CLKIN1；CLKIN_SEL 为 1 时，选择 CLKIN2。参考时钟在 CLKIN1 和 CLKIN2 之间动态切换，切换后需要对 PLL 复位。

对于 GPLL 输入时钟需要注意以下几点：

- 当输入时钟丢失或者不稳定时，需要输入时钟恢复稳定后对 GPLL 复位
- 上电以及上电后无参考时钟，GPLL 会输出一个低频时钟信号，此时 LOCK 信号为低电平，如果想让输出时钟为 0，可以使用关电模式或者时钟输出门控，详见 2.8.6.3 和 2.8.6.12 小节

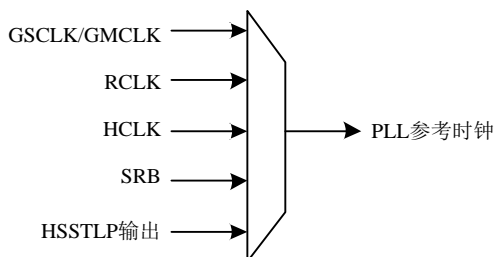


图 2-36 GTP_GPLL 参考时钟源

2.8.6.2 GPLL 复位

RST 复位信号释放之后，PLL 开始进入锁定状态，经过 T_{LOCK} 时间之后完成 PLL 频率锁定。LOCK 信号从低电平到变成高电平之前 CLKOUT 会从低电平逐渐开始翻转，此时 CLKOUT 是不稳定的，LOCK 信号变成高电平后 CLKOUT 才保持稳定。RST 复位信号的时序如下图所示，高电平有效。

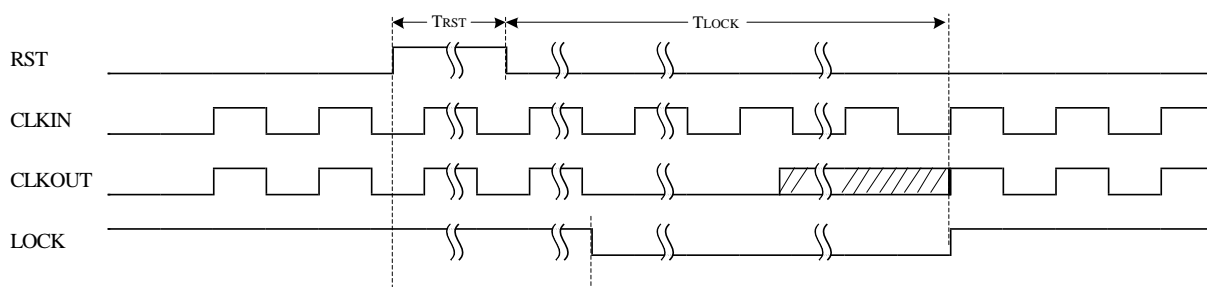


图 2-37 GTP_GPLL 复位时序

T_{RST} 和 T_{LOCK} 性能指标见《DS04001_Logos2 系列 FPGA 器件数据手册》。

2.8.6.3 GPLL 关电

当 PLL 不使用时，可通过将 PLL_PWD 信号置为高电平关闭 PLL 电源，以便节省功耗。如果要再次使用 PLL，将 PLL_PWD 信号置为低电平，经过 T_{LOCK} 时间后 PLL 重新锁定。LOCK 信号从低电平到变成高电平之前 CLKOUT 会从低电平逐渐开始翻转，此时 CLKOUT 是不稳定的，LOCK 信号变成高电平后 CLKOUT 才保持稳定。

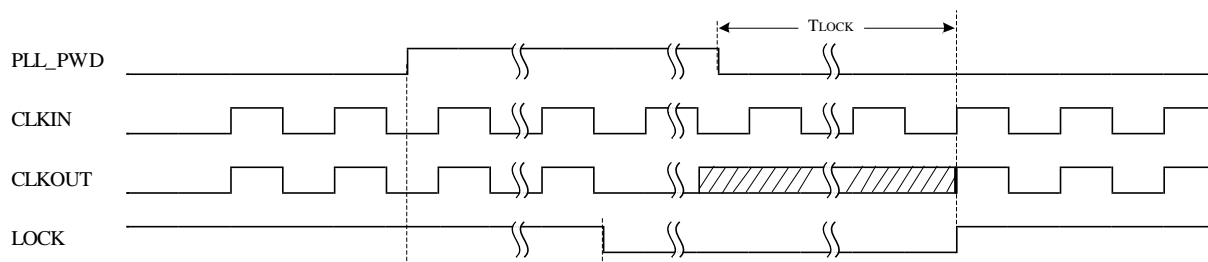


图 2-38 GTP_GPLL 关电时序

2.8.6.4 反馈时钟选择

GPLL 的反馈时钟输入源可以分为 GPLL 内部反馈和 GPLL 外部反馈两种，其中反馈源支持 CLKOUT0~ CLKOUT6,CLKOUTF。

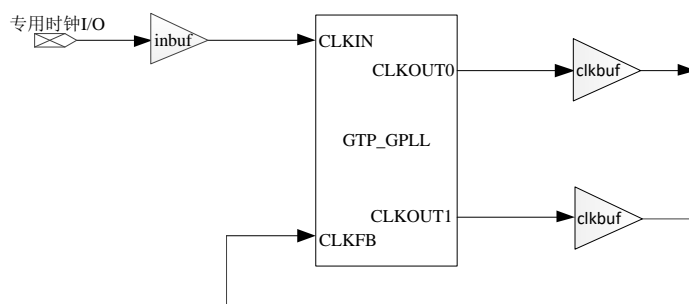


图 2-39 GTP_GPLL 外部反馈示意图

如图所示，选择外反馈时，反馈时钟通过全局时钟或者水平时钟 buffer 走专用时钟路径到达反馈时钟输入，而内反馈模式下，反馈时钟输出在 PLL 内部直接到达反馈时钟输入，相对外反馈受干扰小。

表 2-24 反馈时钟选择

INTERNAL_FB	EXTERNAL_FB	反馈模式
“CLKOUT0”	“DISABLE”	内部反馈模式，选择输出 divider0；
“CLKOUT1”	“DISABLE”	内部反馈模式，选择输出 divider1；
“CLKOUT2”	“DISABLE”	内部反馈模式，选择输出 divider2；
“CLKOUT3”	“DISABLE”	内部反馈模式，选择输出 divider3；
“CLKOUT4”	“DISABLE”	内部反馈模式，选择输出 divider4；
“CLKOUT5”	“DISABLE”	内部反馈模式，选择输出 divider5；
“CLKOUT6”	“DISABLE”	内部反馈模式，选择输出 divider6；
“CLKOUTF”	“DISABLE”	内部反馈模式，选择反馈 F divider；
“DISABLE”	“CLKOUT0”	外部反馈模式，选择输出 divider0；
“DISABLE”	“CLKOUT1”	外部反馈模式，选择输出 divider1；
“DISABLE”	“CLKOUT2”	外部反馈模式，选择输出 divider2；
“DISABLE”	“CLKOUT3”	外部反馈模式，选择输出 divider3；
“DISABLE”	“CLKOUT4”	外部反馈模式，选择输出 divider4；
“DISABLE”	“CLKOUT5”	外部反馈模式，选择输出 divider5；
“DISABLE”	“CLKOUT6”	外部反馈模式，选择输出 divider6；
“DISABLE”	“CLKOUTF”	外部反馈模式，选择反馈 F divider；

2.8.6.5 输出频率编程

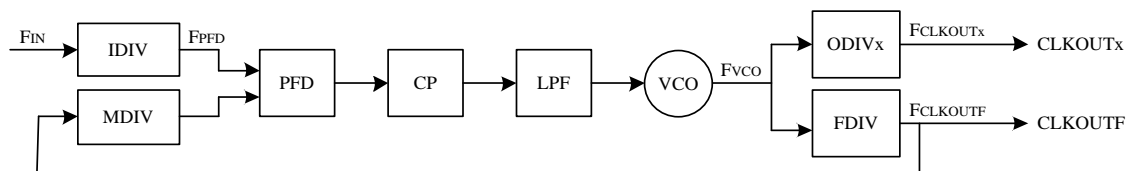


图 2-40 GTP_GPLL 输出频率路径示意图

参考时钟（Fin）经过可编程输入分频器（IDIV）得到鉴频鉴相器（PFD）的一个参考时钟，反馈时钟经过可编程反馈分频器（FDIV）和（MDIV）得到 PFD 另一个参考时钟，PFD 比较这两个时钟的相位和频率，然后产生一个信号驱动电荷泵（CP），CP 产生的电流信号经过环路滤波器（LPF），LPF 产生一个参考电压给压控振荡器（VCO）；PFD 给 CP 和 LPF 提供相位超前或滞后的信号来不断调整 VCO 工作的频率，以此调节频率和相位。例如当 VCO 经过反馈分频器后仍高于参考频率，PFD 产生一个向下调节的信号，使得 CP 和 LPF 减小电压控制 VCO 降低输出频率，当 VCO 经过反馈分频器后仍低于参考频率，PFD 产生一个向上调节的信号，使得 CP 和 LPF 提高电压控制 VCO 增加输出频率。VCO 产生 8 相位输出时钟，通过 8 相位时钟可产生 1 路可变相移时钟，以此实现相移粗调、相位细调和插值相移，每一

个输出相移时钟都可以作为 FDIV 的输入，共有 8 个输出分频器，7 个 ODIV 和一个 FDIV，每个分频器都是独立的，VCO 输出的时钟经过这些分频器得到最终的输出时钟频率。

表 2-25 输出频率编程参数说明

参数名称	功能描述
F_{IN}	输入时钟频率
F_{PFD}	PFD 输入参考时钟频率
F_{VCO}	VCO 输出时钟频率
$F_{CLKOUTx}$	CLKOUT0~6 输出时钟频率
$F_{CLKOUTF}$	CLKOUTF 输出时钟频率
IDIV	输入分频器分频比
MDIV	反馈分频器 M 分频比
ODIV _x	输出时钟 CLKOUT0~6 分频器分频比
FDIV	反馈分频器 F 分频比

注：IDIV 对应参数 STATIC_RATIOI，MDIV 对应参数 STATIC_RATIO_M，ODIV_{0~6} 对应参数 STATIC_RATIO0~6，FDIV 对应参数 STATIC_RATIOF

若内部反馈或外部反馈选择的反馈分频器分频比为 FBDIV，各时钟频率计算公式如下：

$$F_{PFD} = \frac{F_{IN}}{IDIV}$$

$$F_{VCO} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV}$$

$$F_{CLKOUTx} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV \times ODIV_x}$$

$$F_{CLKFBOUT} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV \times FDIV}$$

其中 FBDIV 为反馈环路选择的分频器分频比， $FBDIV \in [ODIV0 \sim 6, FDIV]$ ，如当反馈时钟为 CLKOUT0 时， $FBDIV = ODIV0$ 。

其中输出时钟 CLKOUT0 和 CLKOUTF 不仅支持整数分频模式，还支持分数分频模式。

(1) ODIV0 分频比

$$ODIV0 = \text{CLKOUT0 整数分频比} + \text{CLKOUT0 分数分频比}$$

对应 ODIV0 的分频比对应参数 STATIC_RATIO0 的值。

(2) FDIV 分频比

$$FDIV = \text{CLKOUTF 整数分频比} + \text{CLKOUTF 分数分频比}$$

FDIV 的分频比对应参数 STATIC_RATIOF 的值。

2.8.6.6 环路带宽

GPLL 环路带宽可配置为低带宽（LOW）、中带宽（OPTIMIZED）和高带宽（HIGH），低带宽和中带宽配置的参数一样。

考虑到实际应用中输入时钟非理想时钟，不同带宽配置对 PLL 影响不同：

- 若期望 PLL 输出时钟有较小 Jitter，则配置为低带宽或中带宽，此时锁定时间长
- 若期望 PLL 锁定时间短，则配置为高带宽，此时输出时钟 Jitter 变差

2.8.6.7 分数分频

ODIV0 和 FDIV 支持整数分频和分数分频两种模式。

- 分数分频精度 0.125
- 分数分频模式不支持静态相移模式的插值相移和动态相移模式
- 分数分频模式占空比不可编程

若分数分频比为 N.f，占空比计算公式如下：

$$\text{DutyCycle} = 50\% + \frac{\alpha}{2 \left(8N + \frac{f}{0.125} \right)}$$

其中 $\frac{f}{0.125} = \text{奇数}$ ， $\alpha=1$ ， $\frac{f}{0.125} = \text{偶数}$ ， $\alpha=0$ ；

例：分频系数整数部分 ODIV0 STATIC VALUE=24，分频系数小数部分 ODIV0 Fraction VALUE=1/8 可以得出 N=24，f=0.125， $\alpha=1$ ， $\text{DutyCycle} = 50\% + 1/2(8*24+1)=50.259\%$ 。

只有输出时钟 CLKOUT0 和 CLKOUTF 支持分数分频，通过配置参数 STATIC_RATIO0 实现对 CLKOUT0 的分数分频配置，通过配置参数 STATIC_RATIOF 实现对 CLKOUTF 的分数分频配置，如想让 CLKOUT0 分频比为 2.125，则将参数 STATIC_RATIO0 的值配置为 2.125 即可，CLKOUTF 配置方式同理，需要注意的是输出时钟频率必须在可配范围内。

2.8.6.8 相位调整

相位调整包含相位粗调、相位细调和插值相移三种方式，相位粗调和相位细调既有静态配置模式又可以通过 APB 接口调整，这两种调整方式每路输出时钟是相互独立的；而插值相移既可以静态配置也可以通过动态配置接口 DPS 接口调整，会影响所有选择该模式的输出时钟，作为反馈时钟的输出时钟不能进行相位调整。

➤ 相位调整方式

• 相位粗调

通过配置参数 `STATIC_CPHASE0/1/2/3/4/5/6/F` 可以实现输出时钟 `CLKOUT0/1/2/3/4/5/6/F` 相位粗调。

相对于输入的 VCO 时钟，输出时钟 `CLKOUT0/1/2/3/4/5/6/F` 相位粗调的步阶是 T_{vco} ，调整范围是 $[0 \sim N_{div}-1] * T_{vco}$ ，调整度数步阶是 $360^\circ / N_{div}$ 。

N_{div} 是输出分频比 `ODIVx` 或反馈分频比 `FDIV`。

• 相位细调

通过配置参数 `STATIC_PHASE0/1/2/3/4/5/6/F` 可以实现输出时钟 `CLKOUT0/1/2/3/4/5/6/F` 相位细调。

相对于输入的 VCO 时钟，输出时钟 `CLKOUT0/1/2/3/4/5/6/F` 相位细调的步阶是 $T_{vco}/8$ ，调整范围是 $(0 \sim 7/8)T_{vco}$ ，调整度数步阶是 $45^\circ / N_{div}$ ，调整范围是 $(0 \sim 45^\circ * 7) / N_{div}$ 。

• 插值相移

通过配置参数 `STATIC_PHASE` 和使能信号 `CLK_DPS0/1/2/3/4/5/6/F_EN` 可实现 `CLKOUT0/1/2/3/4/5/6/F` 的静态插值相移，所有输出时钟共用参数 `STATIC_PHASE`，如果需配置 `CLKOUT0` 静态插值相移功能，需要将 `CLK_DPS0_EN` 配置为“TRUE”，并且设置 `STATIC_PHASE` 的值。

通过使能信号 `CLK_DPS0/1/2/3/4/5/6/F_EN` 和 `DPS` 端口可实现 `CLKOUT0/1/2/3/4/5/6/F` 的动态插值相移功能，所有输出时钟共用一个 `DPS` 接口，如果需配置 `CLKOUT0` 静态插值相移功能，需要将 `CLK_DPS0_EN` 配置为“TRUE”，然后通过 `DPS` 接口动态调整 `CLKOUT0` 输出时钟相位。

相对于输入的 VCO 时钟，输出 `CLKOUT0/1/2/3/4/5/6/F` 插值相移精度 $T_{vco}/64$ ，调整度数步阶是 $360^\circ / (64 * N_{div})$ 。

动态调相采用插值相移，输出时钟可以实现 360 度循环旋转，详见本小节“[动态调整](#)”。

注：

1. $N_{div} \in \{STATIC_RATIO0 \sim 6, STATIC_RATIOF\}$ ，是输出时钟 `CLKOUT0/1/2/3/4/5/6/F` 的分频系数

2. 在调相过程中，只能选择相位细调和插值相移中一种方式与相位粗调配合，实现输出时钟相位调整

➤ 相位调整模式

• 静态配置

输出时钟 CLKOUT0/1/2/3/4/5/6/F 相位细调静态配置分别由 STATIC_PHASE0/1/2/3/4/5/6/F 控制。

输出时钟 CLKOUT0/1/2/3/4/5/6/F 相位粗调静态配置分别由 STATIC_CPHASE0/1/2/3/4/5/6/F 控制。

输出时钟 CLKOUT0/1/2/3/4/5/6/F 插值相移静态配置由 STATIC_PHASE 控制。

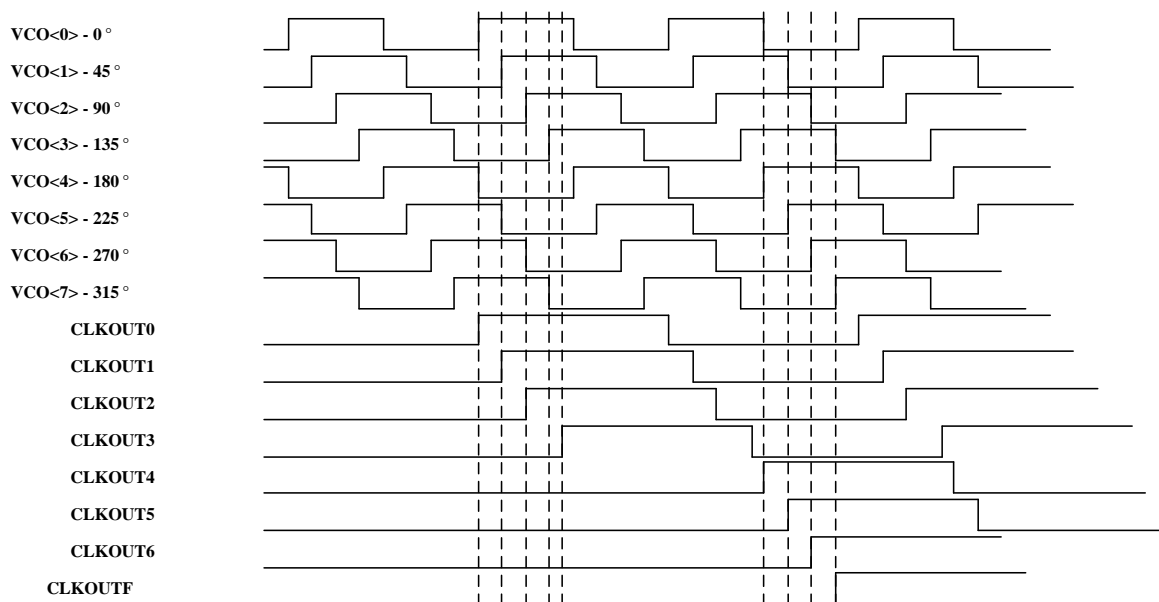


图 2-41 GTP_GPLL 相位调整_静态配置

如下表所示，CLKOUT0 由 VCO<0>二分频产生，细调相位为 0，粗调相位为 0，以 CLKOUT0 为基准，各个输出时钟相位关系如下：

表 2-26 输出时钟相位关系

输出时钟	细调相位	粗调相位	插值相移	相位关系
CLKOUT0	0	0	---	0
CLKOUT1	$1 \cdot T_{vco}/8$	0	---	22.5°
CLKOUT2	$2 \cdot T_{vco}/8$	0	---	45°
CLKOUT3	0	0	$28 \cdot T_{vco}/64$	78.75°
CLKOUT4	$4 \cdot T_{vco}/8$	T_{vco}	---	270°
CLKOUT5	$5 \cdot T_{vco}/8$	T_{vco}	---	292.5°
CLKOUT6	$6 \cdot T_{vco}/8$	T_{vco}	---	315°
CLKOUTF	$7 \cdot T_{vco}/8$	T_{vco}	---	337.5°

• 动态调整

LOGOS2 系列器件的 GTP_GPLL 模块支持相位动态调整，动态相移调整的接口。动态相

位调整可以使得 PLL 的输出时钟相位相对于参考时钟动态的调整。动态相位调整功能通过 DPS_CLK、DPS_EN、DPS_DIR 和 DPS_DONE 四个信号实现。可以通过设置 DPS_DIR 实现相位的增大或者减小，每一次调整，GPLL 的输出时钟相位变化 $1/64 \cdot T_{vco}$ 。相位调整没有最大值，任何时钟频率可以实现 360° 的相位调整，当一个周期结束后，相位调整再次循环开始。

动态相位调整过程如下图所示，DPS_EN、DPS_DIR 和 DPS_DONE 被同步到 DPS_CLK 时钟域。DPS_EN 为高电平时（一个 DPS_CLK 时钟周期），相位调整初始化，DPS_DIR 为低电平时增大相位，而 DPS_DIR 为高电平时减小相位，当相位调整完成时 DPS_DONE 持续一个 DPS_CLK 周期高电平。整个调整过程持续 15 个钟周期。

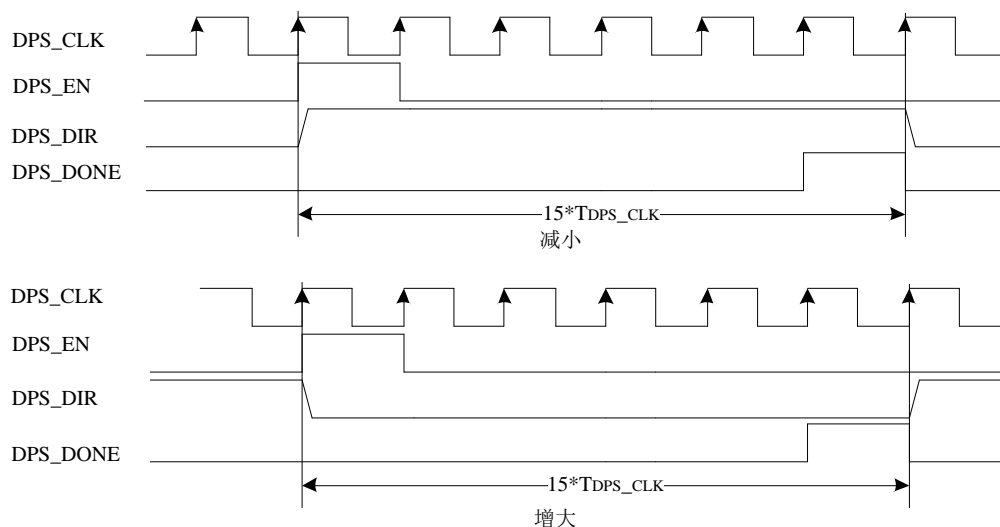


图 2-42 GTP_GPLL 动态相移模式接口时序

动态相移接口如下表：

表 2-27 动态相移端口说明

序号	端口名称	I/O	描述	初始值
1	DPS_CLK	I	动态相位调整时钟信号；	
2	DPS_EN	I	动态相位调整使能信号；	
3	DPS_DIR	I	动态相位调整方向控制； 1'b0: 增大, 1'b1: 减小；	
4	DPS_DONE	O	动态相位调整完成指示信号；	1'b0

注：DPS_CLK 性能指标见《DS04001_Logos2 系列FPGA 器件数据手册》

2.8.6.9 占空比可编程

输出 divider 支持可编程的 Duty Cycle，最小 step 为 50%/odiv，odiv 为输出时钟 divider 分频系数。STATIC_DUTY0/1/2/3/4/5/6/F 决定了分频后 duty cycle，表达式如下，duty 为 duty

cycle 配置值。

$$\text{duty cycle} = (50\% / \text{odiv}) * \text{duty}。$$

例如 odiv 为 24，则根据公式 duty 可配范围为 2~47，占空比范围为 4.17%~97.92%。

注意：

- 1.DUTY 的设置值受到 odiv 值（输出分频）限制：odiv>1 时， $2 \leq \text{duty} \leq 2 * \text{odiv} - 1$ ；odiv=1，duty 无效，不起作用，默认 50% 输出。
- 2.分数分频不支持占空比编程。

2.8.6.10 扩频时钟产生和输入

- 支持频谱扩展时钟输入，调制频率小于 GPLL 带宽
- 支持本地扩频时钟产生，支持中心扩频和向下扩频两种模式
- 中心扩频 HIGH 最大 $\pm 2\%$ ，中心扩频 LOW 最大 $\pm 1\%$ ，向下扩频 HIGH 最大 -2%；向下扩频 LOW 最大 -1%

中心扩频如下图所示，其中幅度较大的实线代表高-中心扩频调制，幅度较低的实线代表低-中心扩频调制。

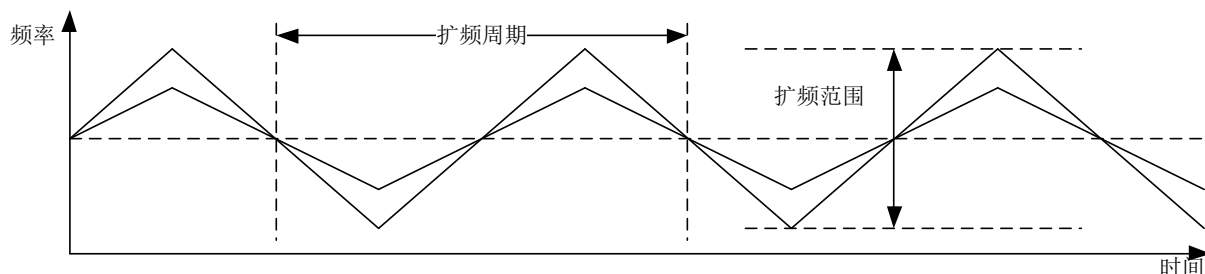


图 2-43 GPLL 中心扩频示意图

向下扩频如下图所示，其中幅度较大的实线代表高-向下扩频调制，幅度较低的实线代表低-向下扩频调制。

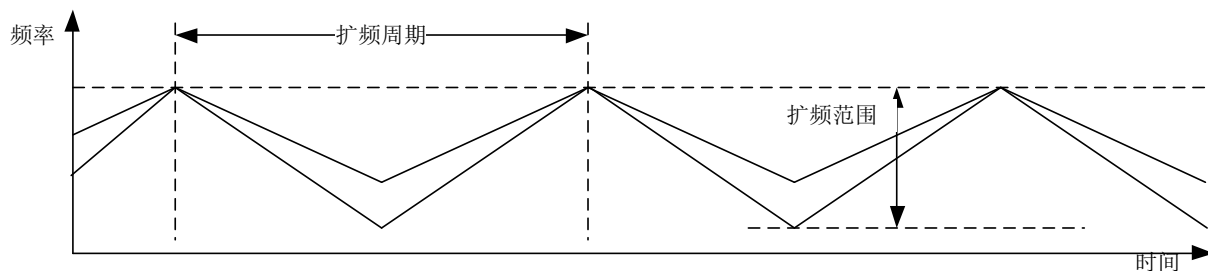


图 2-44 GPLL 向下扩频示意图

GPLL 工作在 SSC 模式下，需要用到原有时钟资源产生扩频时钟，因此：

- CLKOUT2/N 和 CLKOUT3/N 不再独立使用
- 动态相移模式不再使用

2.8.6.11 时钟输出级联

当 CLK_CAS5_EN 配置为“TRUE”时，输出 divider5 和输出 divider6 可以实现时钟级联。

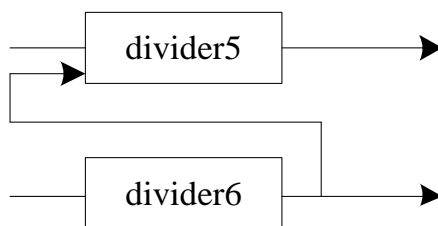


图 2-45 GTP_GPLL OUTPUT Divider 输出级联

级联后输出时钟 CLKOUT5 与其它输出时钟存在固定相位偏差。

2.8.6.12 时钟输出门控

GTP_GPLL 输入信号 CLKOUT0_SYN 用来使能和关闭 CLKOUT0 输出时钟，当用户不需要 CLKOUT0 时钟时，可以关闭该时钟以节省 Clock Tree 和 Fabric 功耗。

CLKOUT0_SYN 信号可选，当用户配置 CLKOUT0_SYN_EN 为“TRUE”时，该信号有效。

同样地，输入信号 CLKOUT1~6/F_SYN 分别与配置位 CLKOUT1~6/F_SYN_EN 配合，使能和关闭 CLKOUT1~6/F 时钟。

以输出时钟 CLKOUT0 为例，CLKOUT0_SYN 电平由 0 跳变到 1 时，经过 3 个 CLKIN 下降沿 CLKOUT0 输出低电平，CLKOUT0_SYN 电平由 1 跳变到 0 时，经过 3 个 CLKIN 下降沿 CLKOUT0 输出有效。功能示意如下图所示：

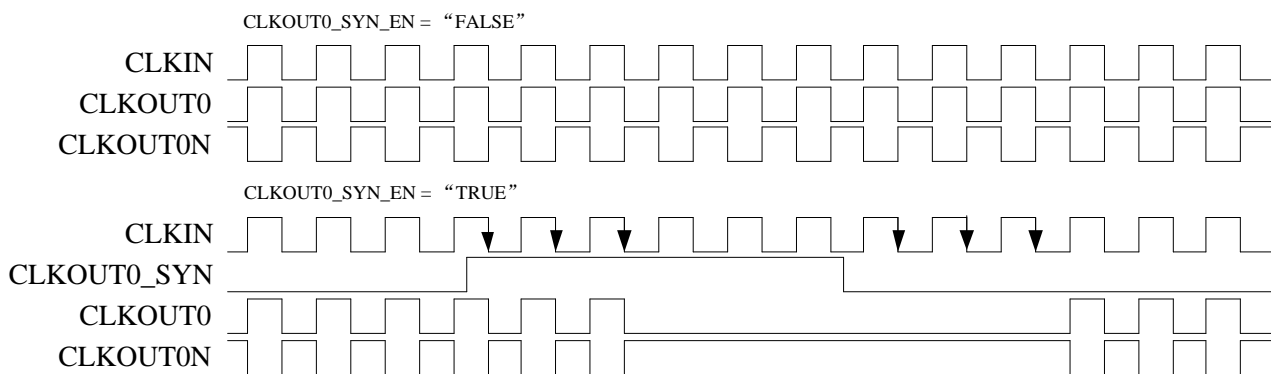


图 2-46 GTP_GPLL 输出时钟 GATE 时序图

2.8.6.13 LOCK 指示

GPLL 支持两种 LOCK 模式，由配置参数 LOCK_MODE 进行选择。

表 2-28 GPLL LOCK 指示模式

参数名	参数值	功能描述
LOCK_MODE	0	实时监测 PLL LOCK 的状态，一旦进入非正常状态则会拉低 LOCK
	1	LOCK 不会实时监控（指示）PLL 工作状态；LOCK 一旦拉高则不会拉低，除非复位或 Power Down

2.8.7 GTP_GPLL 配置约束

在实际应用中，要实现某些功能的时候需要配置相关参数，这些参数的可配范围互相影响，当某个参数确定下来的时候其它参数的可配范围就会受到限制，而不是在这些参数的整个范围内都是可配，如果参数配错就可能导致 GPLL 工作异常，所以在配置的时候一定要知道相关参数的限制条件和可配范围。

可以通过两种方式来配置 GPLL，一种是通过 GUI（开发工具中 IP compiler）实现，关于这种配置的使用方法参考《UG041001_28nm_PLL_IP_UserGuide》，在 IP 配置界面可以找到；另一种方式就是例化原语，然后根据具体实现的功能去配置相关参数，可以结合 IP compiler 中 GPLL Advanced 模式去了解配置约束，比如某些参数设置范围不对，IP 生成时会报错，同时也会提醒可配范围。

参数可配范围请详见表 2-21 和《DS04001_Logos2 系列FPGA 器件数据手册》。

输出时钟频率约束：

GPLL 最常用的功能之一就是配置期望的输出频率，以下因素需要考虑：

- 选择输入频率必须在可配范围内

- 输出时钟频率必须在可配范围内
- PFD 频率必须在可配范围内
- 输入分频参数 IDIV、输出分频参数 ODIV，反馈分频参数 MDIV、FBDIV 必须在可配范围内
- VCO 频率范围必须在规定范围内

假如期望的输出时钟频率是 100MHz，输入时钟频率选择的是 50MHz；

首先考虑 PFD 频率，根据公式，由于输入时钟是 50MHz，PFD 频率范围是 10MHz 到 450MHz，所以 IDIV 可配范围 1 到 5，假如选择 1，选择内反馈模式，反馈输出时钟频率是 50MHz，保持与输入时钟频率一致，此时 FBDIV=FDIV，根据公式得到 MDIV=1。

由以上两步确定的参数：CLKIN1=50MHz，PFD 频率=50MHz，CLKOUT0=100MHz，IDIV=MDIV=1。

接着根据 VCO 公式和 VCO 频率范围得到 FBDIV 的可配范围为 12 到 24，如果选择 24，则根据输出时钟频率公式得到 ODIV0=6，VCO 频率是 1200MHz。

通过以上步骤我们就得到了一个有效的输出时钟频率。

2.8.8 GTP_GPLL 应用模式

2.8.8.1 内部反馈模式

内部反馈模式主要针对某些应用场景下，对 GPLL 输出时钟具有高性能的要求，比如频率综合、低抖动时钟输出等。

内部反馈模式下，反馈时钟直接连接到 GPLL 的输入端，不经过任何其他电路，这样就免除了其他电路引入的噪声，拥有更好的抖动性能。

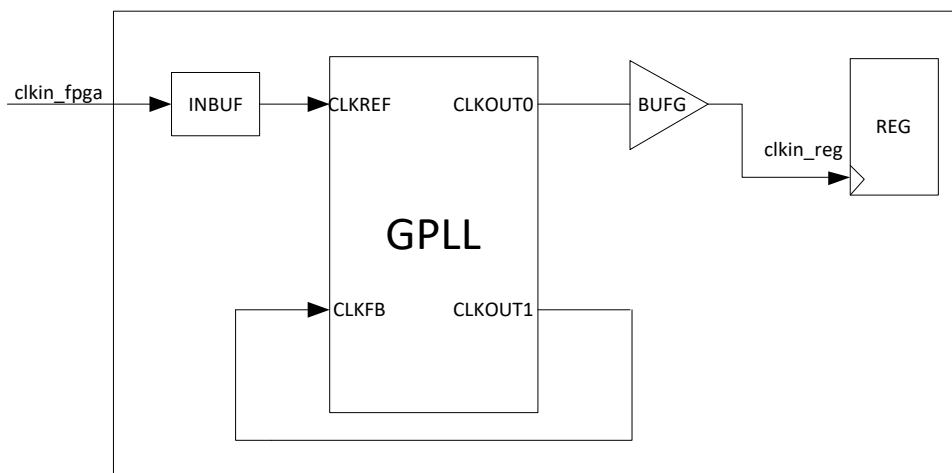


图 2-47 GPLL 内部反馈模式示意图

2.8.8.2 源同步模式

采用外部反馈实现源同步模式，源同步模式主要针对 GPLL 输出时钟驱动内部寄存器时，需要保证输入时钟与数据保持一定的相位关系，如下图所示。GPLL 锁定时，b/e 两点相位是对齐的，源同步模式下，GPLL 输出时钟 clkout0 送给寄存器的路径与反馈时钟 clkout1 所经历的路径，必须对称平衡，且使用同一 regional buffer，也就是保证 c 点与 e 点时钟相位对齐，即 b 点与 c 点时钟相位对齐；同时，可以通过 GPLL 相位调整功能补偿数据管脚到寄存器数据端输入和时钟管脚到 GPLL 输入参考时钟之间的 skew(偏移)，即 d 点到 f 点和 a 点到 b 点的偏移，这样可以在 FPGA 内部 REG 处，可以实现数据与时钟之间相位边沿或中心对齐，具体是 0° 还是 90° 对齐，取决于用户的要求，具体的相移要求可以通过输出时钟的分频器的 phase shift 功能予以实现。

例如，输入管脚时钟上升沿处于数据的中央进入 FPGA，可以采用 GPLL 输出 phase shift 功能，移动 clkout0 的相位，在 FPGA 内部的寄存器，可以实现时钟上升沿对准数据的中央。此时，如果输入时钟 clkin_fpga 进入 GPLL 的参考输入与 data 进入 register 之间如没有任何 skew，比如走专用时钟路径，即 skew 很小，此时，GPLL 可以不用任何 phase shift 调整，自动实现上述的时钟上升沿与数据中央中心对齐。

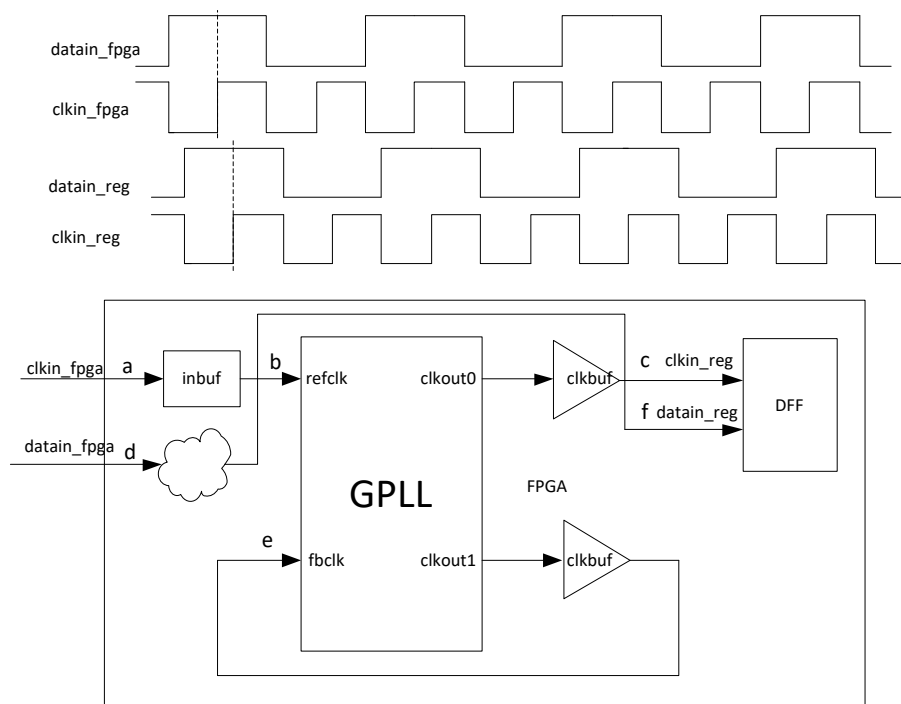


图 2-48 GPLL 源同步模式示意图

2.8.8.3 零延迟模式

采用外部管脚反馈实现零延迟模式，主要针对只有一个输入参考时钟情况下，且 FPGA 驱动多个外部器件情形时，需要保证外部器件的输入时钟与 FPGA 输入管脚的时钟对齐，如下图所示。

零延迟模式下，输入时钟与反馈时钟需要使用相同的 IO 标准，以保证 a 点与 d 点时钟相位对齐，b 点与 e 点相位对齐。同时，板上反馈时钟 trace 与驱动外部器件的时钟 trace 相互匹配等长，则 d 点与 c 点时钟相位对齐，于是 a、c 点时钟对齐，实现零延迟缓冲的功能。

实际情况下，由于外部器件的负载寄生电容与 FPGA 输入管脚的寄生电容不一致，会导致 c 点与 d 点的信号上升沿或下降沿时间不一致，会导致延迟略有差异。因此这个 a/c 时钟 skew 的差异，客户需要在时序收敛时必须予以考虑，留有一定的余量。

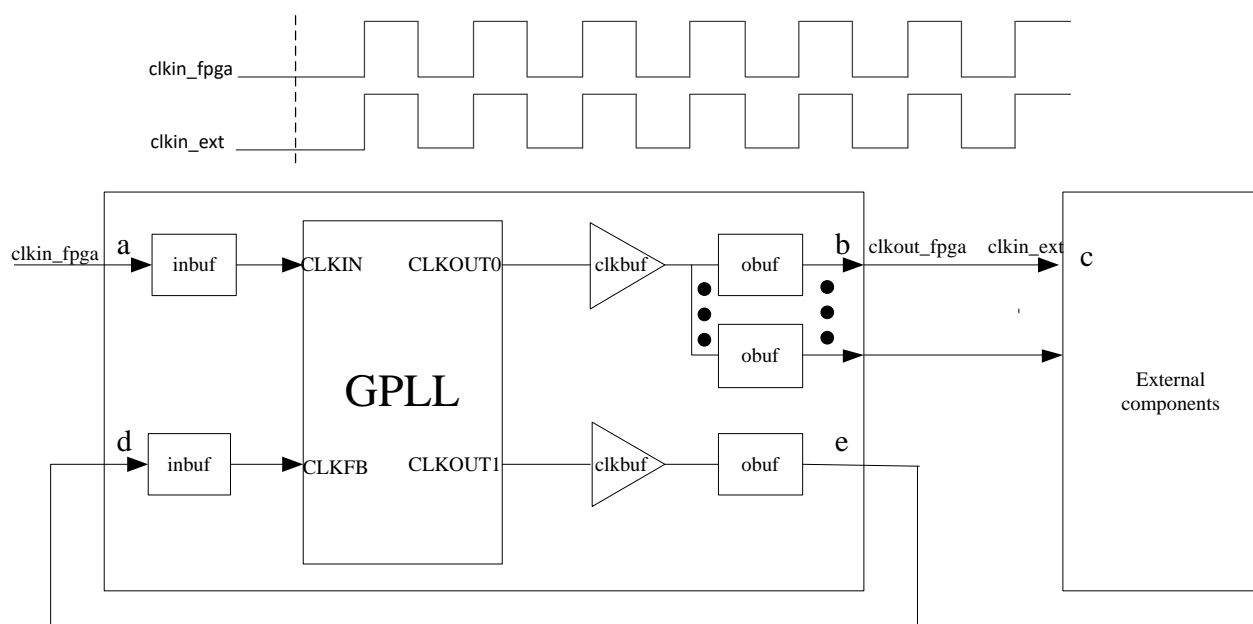


图 2-49 GPLL 零延迟模式示意图

2.9 PPLL 使用说明

2.9.1 PPLL 概述

PPLL 是 FPGA 提供时钟资源的核心子系统，PGL2G100H 内部有 6 个 PPLL 模块。PPLL 主要给 FPGA 提供：时钟频率综合，降低时钟偏移(skew)，相位调整、输出时钟 Gate 以及 APB 动态重配等功能。

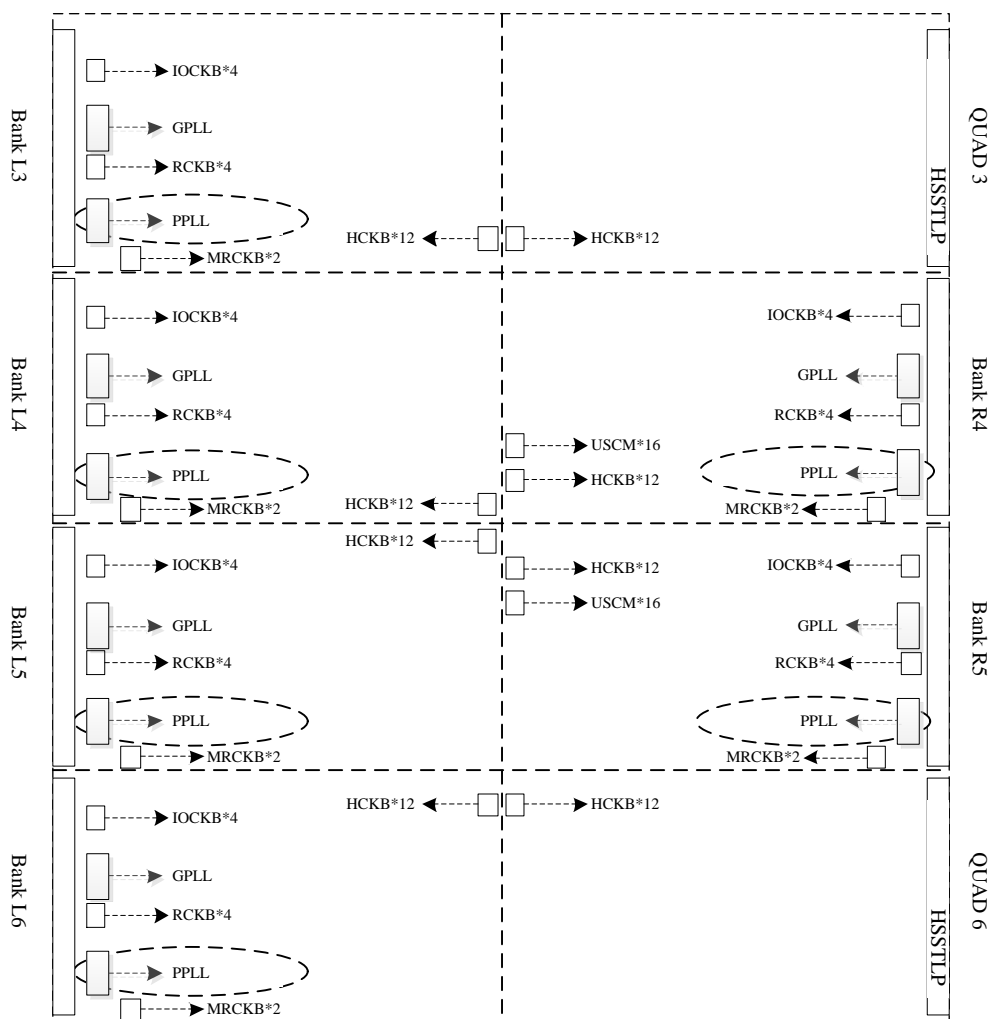


图 2-50 PPLL 资源示意图

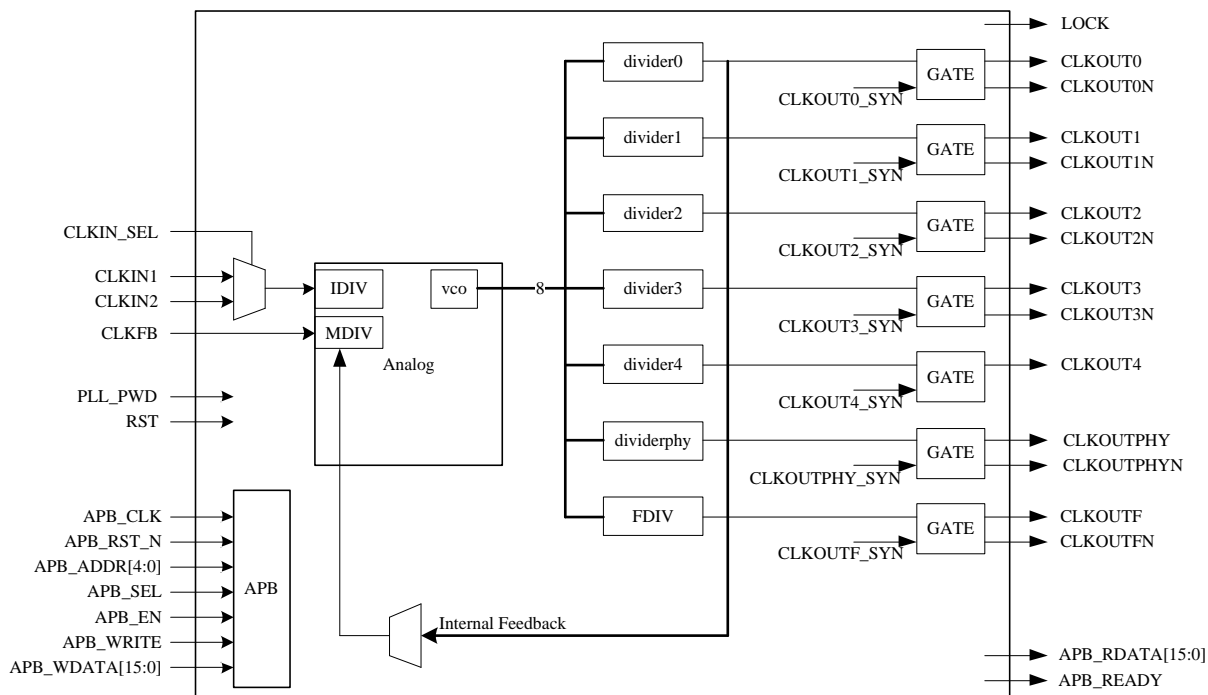


图 2-51 PPLL 系统框图

在使用中通过调用 GTP_PPLL 实例配置使用，对于不用的时钟输出端口可以配置输出关断来降低功耗。在 GTP_PPLL 的 CLKOUTPHY 的时钟输出专用于 DDR PHY 的时钟，用户逻辑不要使用。下面介绍 GTP_PPLL 的应用。

2.9.2 GTP_PPLL 顶层框图

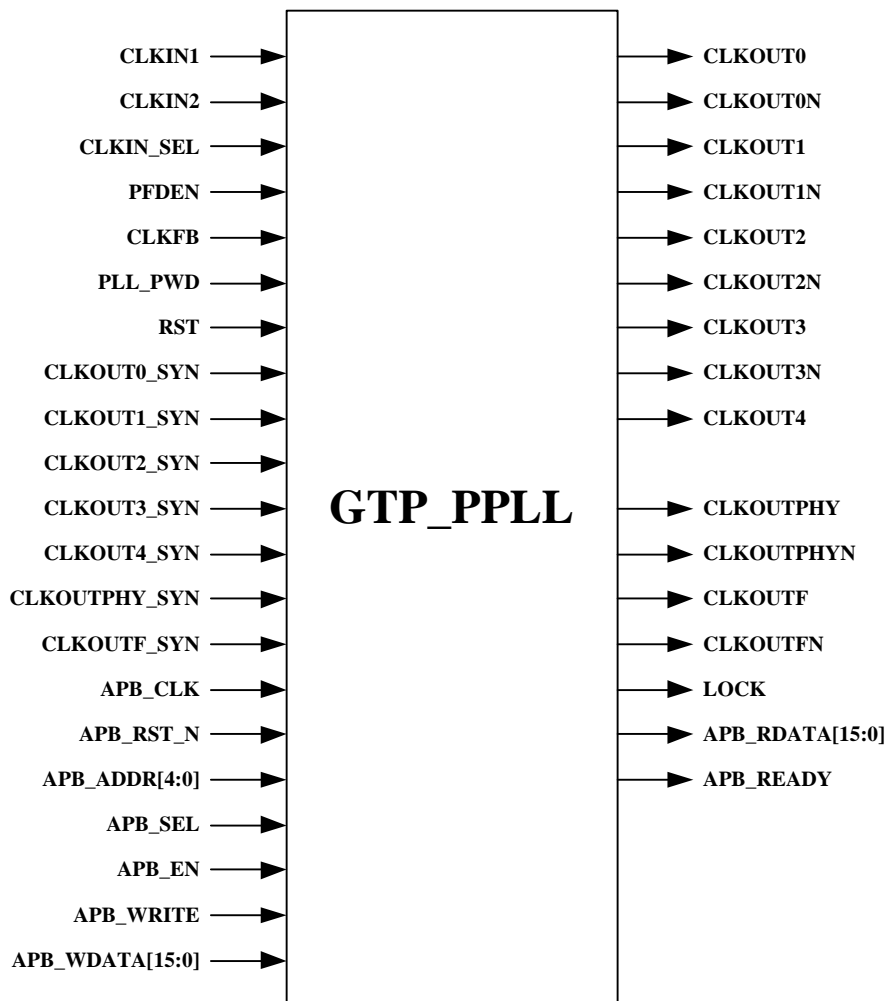


图 2-52 GTP_PPLL 框图

2.9.3 GTP_PPLL 端口列表

表 2-29 GTP_PPLL 端口列表

Port	I/O	功能描述
CLKOUT0	O	PLL 第 0 路正相输出时钟;
CLKOUT0N	O	PLL 第 0 路反相输出时钟;
CLKOUT1	O	PLL 第 1 路正相输出时钟;
CLKOUT1N	O	PLL 第 1 路反相输出时钟;
CLKOUT2	O	PLL 第 2 路正相输出时钟;
CLKOUT2N	O	PLL 第 2 路反相输出时钟;
CLKOUT3	O	PLL 第 3 路正相输出时钟;
CLKOUT3N	O	PLL 第 3 路反相输出时钟;
CLKOUT4	O	PLL 第 4 路正相输出时钟;
CLKOUTPHY	O	PLL 给 DDR PHY 正相输出时钟;

Port	I/O	功能描述
CLKOUTPHYN	O	PLL 给 DDR PHY 反相输出时钟；
CLKOUTF	O	PLL 反馈正相输出时钟；
CLKOUTFN	O	PLL 反馈反相输出时钟；
LOCK	O	PLL 频率锁定指示信号，为异步信号； 信号拉高时，指示 PLL 反馈时钟信号已锁定到输入时钟信号；
APB_RDATA[15:0]	O	PLL APB 接口数据总线数据输出；
APB_READY	O	PLL APB 接口数据总线握手信号；表示一个正常的总线周期结束；
CLKIN1	I	PLL 参考输入时钟 1；
CLKIN2	I	PLL 参考输入时钟 2；
CLKFB	I	PLL 反馈时钟；
CLKIN_SEL	I	输入时钟选择信号；
PLL_PWD	I	PLL Power Down；高电平有效；
RST	I	PLL 复位信号，高电平有效；
CLKOUT0_SYN	I	CLKOUT0/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT1_SYN	I	CLKOUT1/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT2_SYN	I	CLKOUT2/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT3_SYN	I	CLKOUT3/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUT4_SYN	I	CLKOUT4 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUTPHY_SYN	I	CLKOUTPHY 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
CLKOUTF_SYN	I	CLKOUTF/N 输出时钟使能控制；高电平有效；高电平关闭，低电平使能；
APB_CLK	I	PLL APB 接口数据总线时钟；
APB_RST_N	I	PLL APB 接口数据总线异步复位信号，低电平有效；
APB_ADDR	I	PLL APB 接口数据总线地址；
APB_SEL	I	PLL APB 接口数据总线选择信号，选择从设备；
APB_EN	I	PLL APB 接口数据总线使能信号；
APB_WRITE	I	PLL APB 接口数据总线写使能信号；1'b0：读操作，1'b1：写操作；
APB_WDATA[15:0]	I	PLL APB 接口数据总线数据输入；

2.9.4 GTP_PPLL 参数列表

表 2-30 GTP_PPLL 参数列表

参数名称	参数类型	有效值	功能描述
CLKIN_FREQ	real	19~800	输入时钟频率配置，MHz；
LOCK_MODE	binary	1'b0,1'b1	PLL 频率检测模式配置； 1'b0：实时监测 PLL 工作状态； 1'b1：PLL 锁定后 LOCK 一直保持，除非复位或 Power Down；
STATIC_RATIOI	interger	1-40	输入 divider ratio 静态配置；
STATIC_RATIO_M	interger	1-128	反馈 M divider ratio 静态配置；
STATIC_RATIOO	interger	1-128	输出 divider0 ratio 静态配置；

参数名称	参数类型	有效值	功能描述
STATIC_RATIO1	interger	1-128	输出 divider1 ratio 静态配置;
STATIC_RATIO2	interger	1-128	输出 divider2 ratio 静态配置;
STATIC_RATIO3	interger	1-128	输出 divider3 ratio 静态配置;
STATIC_RATIO4	interger	1-128	输出 divider4 ratio 静态配置;
STATIC_RATIOPHY	interger	1-128	输出 dividerphy ratio 静态配置;
STATIC_RATIOF	interger	1-128	反馈 F divider ratio 静态配置;
STATIC_DUTY0	interger	2~255	输出 divider0 duty 静态配置;
STATIC_DUTY1	interger	2~255	输出 divider1 duty 静态配置;
STATIC_DUTY2	interger	2~255	输出 divider2 duty 静态配置;
STATIC_DUTY3	interger	2~255	输出 divider3 duty 静态配置;
STATIC_DUTY4	interger	2~255	输出 divider4 duty 静态配置;
STATIC_DUTYPHY	interger	2~255	输出 dividerphy duty 静态配置;
STATIC_DUTYF	interger	2~255	反馈 F divider duty 静态配置;
STATIC_PHASE0	interger	0-7	输出 divider0 fine phase 静态配置;
STATIC_PHASE1	interger	0-7	输出 divider1 fine phase 静态配置;
STATIC_PHASE2	interger	0-7	输出 divider2 fine phase 静态配置;
STATIC_PHASE3	interger	0-7	输出 divider3 fine phase 静态配置;
STATIC_PHASE4	interger	0-7	输出 divider4 fine phase 静态配置;
STATIC_PHASEPHY	interger	0-7	输出 dividerphy fine phase 静态配置;
STATIC_PHASEF	interger	0-7	反馈 F divider fine phase 静态配置;
STATIC_CPHASE0	interger	0-127	输出 divider0 coarse phase 静态配置;
STATIC_CPHASE1	interger	0-127	输出 divider1 coarse phase 静态配置;
STATIC_CPHASE2	interger	0-127	输出 divider2 coarse phase 静态配置;
STATIC_CPHASE3	interger	0-127	输出 divider3 coarse phase 静态配置;
STATIC_CPHASE4	interger	0-127	输出 divider4 coarse phase 静态配置;
STATIC_CPHASEPHY	interger	0-127	输出 dividerphy coarse phase 静态配置;
STATIC_CPHASEF	interger	0-127	反馈 F divider coarse phase 静态配置;
CLKOUT0_SYN_EN	string	“FALSE”,“TRUE”	CLKOUT0_SYN 信号使能配置;
CLKOUT1_SYN_EN	string	“FALSE”,“TRUE”	CLKOUT1_SYN 信号使能配置;
CLKOUT2_SYN_EN	string	“FALSE”,“TRUE”	CLKOUT2_SYN 信号使能配置;
CLKOUT3_SYN_EN	string	“FALSE”,“TRUE”	CLKOUT3_SYN 信号使能配置;
CLKOUT4_SYN_EN	string	“FALSE”,“TRUE”	CLKOUT4_SYN 信号使能配置;
CLKOUTPHY_SYN_EN	string	“FALSE”,“TRUE”	CLKOUTPHY_SYN 信号使能配置;
CLKOUTF_SYN_EN	string	“FALSE”,“TRUE”	CLKOUTF_SYN 信号使能配置;

参数名称	参数类型	有效值	功能描述
INTERNAL_FB	string	“CLKOUT0”, “CLKOUT1”, “CLKOUT2”, “CLKOUT3”, “CLKOUT4”, “CLKOUTF”, “DISABLE”	内部反馈路径选择，选择这几个输出时钟其中一个作为内反馈时钟，如 INTERNAL_FB = " CLKOUT0", 则输出时钟 CLKOUT0 作为内反馈时钟；
EXTERNAL_FB	string	“CLKOUT0”, “CLKOUT1”, “CLKOUT2”, “CLKOUT3”, “CLKOUT4”, “CLKOUTF”, “DISABLE”	外部反馈路径选择，选择这几个输出时钟其中一个作为外反馈时钟如 EXTERNAL_FB = " CLKOUT0", 则输出时钟 CLKOUT0 作为外反馈时钟；
BANDWIDTH	string	“LOW”, “HIGH” “OPTIMIZED”	带宽选择配置： BANDWIDTH=“LOW”配置为“LOW”， BANDWIDTH=“HIGH”配置为“HIGH” BANDWIDTH=“OPTIMIZED”配置为“OPTIMIZED”；

注:

1. 参数 INTERNAL_FB 和参数 EXTERNAL_FB 不能同时配置为“DISABLE”
2. BANDWIDTH 配置受到 PFD 频率限制

2.9.5 GTP_PPLL APB 接口配置

用户逻辑通过 APB 接口对 GTP_PPLL 工作参数进行动态地改变。APB 接口读写寄存器时序如下图所示：

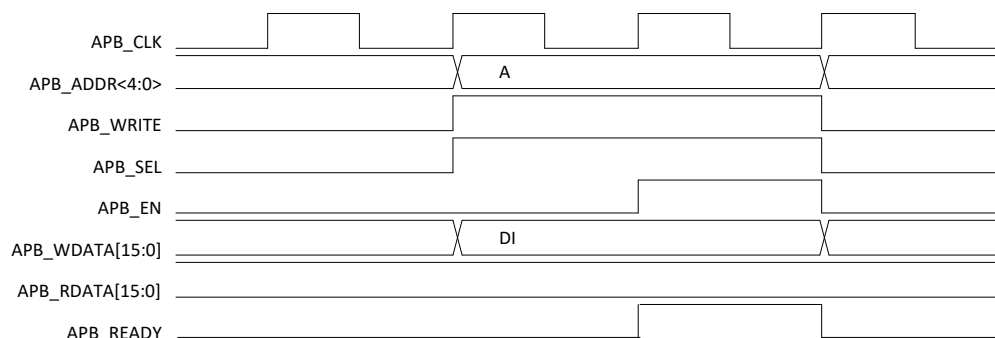


图 2-53 GTP_PPLL_APB 写时序

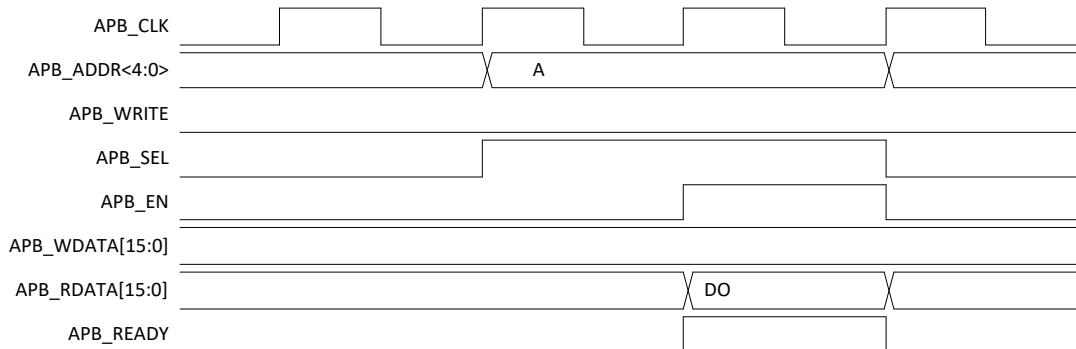


图 2-54 GTP_PPLL_APB 读时序

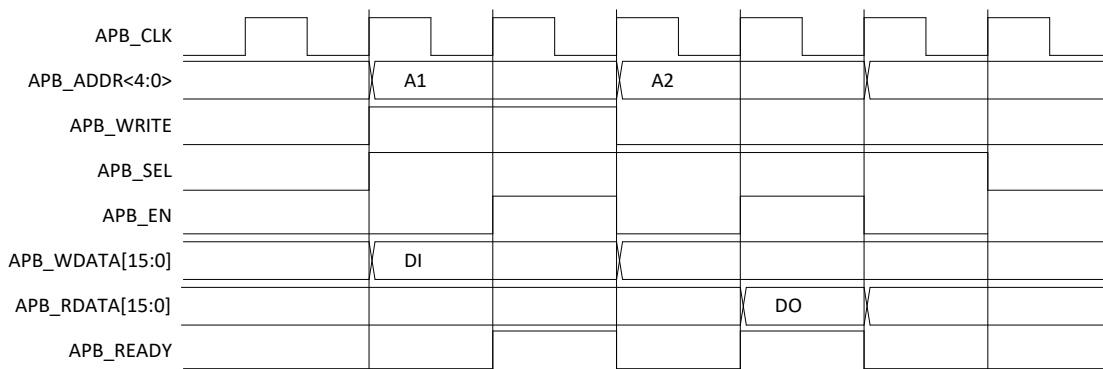


图 2-55 GTP_PPLL_APB 写后读时序

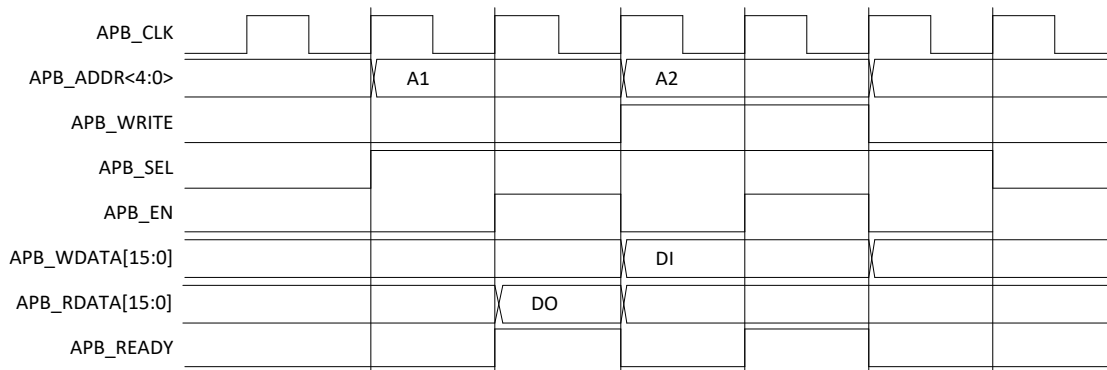


图 2-56 GTP_PPLL_APB 接口读后写时序

表 2-31 GTP_PPLL_APB 接口端口列表

序号	端口名称	I/O	描述
1	APB_CLK	I	时钟，上升沿采样；
2	APB_RST_N	I	异步复位，低有效；只复位 Data Bus（不复位寄存器）；
3	APB_ADDR[4:0]	I	地址总线；
4	APB_SEL	I	选择，表示从设备被选择；
5	APB_EN	I	使能，总线使能信号；
6	APB_WRITE	I	方向，0：读，1：写；
7	APB_WDATA[15:0]	I	数据总线输入；
8	APB_RDATA[15:0]	O	数据总线输出；

序号	端口名称	I/O	描述
9	APB_READY	O	标志信号；

通过 APB 接口写内部寄存器的系统应用要求：

- APB 改写内部寄存器后，需要复位 PLL，保证其正常工作
- RST 最小脉冲宽度性能指标见 《DS04001_Logos2 系列 FPGA 器件数据手册》

表 2-32 GTP_PPLL_APB 接口寄存器列表

Address	Bit	寄存器	描述
0	[15:8]	REG_ODIV0_RATIO<7:0>	ODIV0 Register1: 分频比设置；
	[7:0]	REG_ODIV0_DUTY<7:0>	ODIV0 Register1: 占空比设置；
1	[15:11]	RESERVED	
	[10]	REG_ODIV0_MUXSEL_EN	ODIV0 Register2: 输入时钟 MUX 使能；
	[9:7]	REG_ODIV0_FPHASE<2:0>	ODIV0 Register2: 相位细调设置；
	[6:0]	REG_ODIV0_CPHASE<6:0>	ODIV0 Register2: 相位粗调设置；
2	[15:8]	REG_ODIV1_RATIO<7:0>	ODIV1 Register1: 分频比设置；
	[7:0]	REG_ODIV1_DUTY<7:0>	ODIV1 Register1: 占空比设置；
3	[15:11]	RESERVED	
	[10]	REG_ODIV1_MUXSEL_EN	ODIV1 Register2: 输入时钟 MUX 使能；
	[9:7]	REG_ODIV1_FPHASE<2:0>	ODIV1 Register2: 相位细调设置；
	[6:0]	REG_ODIV1_CPHASE<6:0>	ODIV1 Register2: 相位粗调设置；
4	[15:8]	REG_ODIV2_RATIO<7:0>	ODIV2 Register1: 分频比设置；
	[7:0]	REG_ODIV2_DUTY<7:0>	ODIV2 Register1: 占空比设置；
5	[15:11]	RESERVED	
	[10]	REG_ODIV2_MUXSEL_EN	ODIV2 Register2: 输入时钟 MUX 使能；
	[9:7]	REG_ODIV2_FPHASE<2:0>	ODIV2 Register2: 相位细调设置；
	[6:0]	REG_ODIV2_CPHASE<6:0>	ODIV2 Register2: 相位粗调设置；
6	[15:8]	REG_ODIV3_RATIO<7:0>	ODIV3 Register1: 分频比设置；
	[7:0]	REG_ODIV3_DUTY<7:0>	ODIV3 Register1: 占空比设置；
7	[15:11]	RESERVED	
	[10]	REG_ODIV3_MUXSEL_EN	ODIV3 Register2: 输入时钟 MUX 使能；
	[9:7]	REG_ODIV3_FPHASE<2:0>	ODIV3 Register2: 相位细调设置；
	[6:0]	REG_ODIV3_CPHASE<6:0>	ODIV3 Register2: 相位粗调设置；
8	[15:8]	REG_ODIV4_RATIO<7:0>	ODIV4 Register1: 分频比设置；
	[7:0]	REG_ODIV4_DUTY<7:0>	ODIV4 Register1: 占空比设置；
9	[15:11]	RESERVED	
	[10]	REG_ODIV4_MUXSEL_EN	ODIV4 Register2: 输入时钟 MUX 使能；
	[9:7]	REG_ODIV4_FPHASE<2:0>	ODIV4 Register2: 相位细调设置；
	[6:0]	REG_ODIV4_CPHASE<6:0>	ODIV4 Register2: 相位粗调设置；
A	[15:8]	REG_ODIVPHY_RATIO<7:0>	ODIVPHY Register1: 分频比设置；

Address	Bit	寄存器	描述
	[7:0]	REG_ODIVPHY_DUTY<7:0>	ODIVPHY Register1: 占空比设置;
B	[15:11]	RESERVED	
	[10]	REG_ODIVPHY_MUXSEL_EN	ODIVPHY Register2: 输入时钟 MUX 使能;
	[9:7]	REG_ODIVPHY_FPHASE<2:0>	ODIVPHY Register2: 相位细调设置;
	[6:0]	REG_ODIVPHY_CPHASE<6:0>	ODIVPHY Register2: 相位粗调设置;
C	[15:8]	REG_FDIV_RATIO<7:0>	FDIV Register1: 分频比设置;
	[7:0]	REG_FDIV_DUTY<7:0>	FDIV Register1: 占空比设置;
D	[15:11]	RESERVED	
	[10]	REG_FDIV_MUXSEL_EN	FDIV Register2: 输入时钟 MUX 使能;
	[9:7]	REG_FDIV_FPHASE<2:0>	FDIV Register2: 相位细调设置;
	[6:0]	REG_FDIV_CPHASE<6:0>	FDIV Register2: 相位粗调设置;
E	[15:7]	RESERVED	
	[6:0]	REG_IDIV_RATIO<7:0>	IDIV Register: 分频比设置;
F	[15:7]	RESERVED	
	[6:0]	REG_MDIV_RATIO<7:0>	MDIV Register: 分频比设置;
10	[15:14]	RESERVED	BANDWIDTH Register
	[13:12]	REG_VCTRL_INIT<1:0>	
	[11:10]	REG_CP_SELFBIAS_SEL<1:0>	
	[9:8]	REG_ICP_BASE_SEL<1:0>	
	[7:4]	REG_CP_CUR_SEL<3:0>	
	[3:1]	REG_LPF_R<2:0>	
	[0]	REG_LPF_C	
11	[15:6]	RESERVED	LOCK Register1
	[5]	REG_LOCK_FILTER_PD	
	[4:0]	REG_FREQ_LOCKDET_SET<4:0>	

2.9.6 GTP_PPLL 功能说明

2.9.6.1 时钟输入

CLKIN_SEL 为 0 时，选择 CLKIN1；CLKIN_SEL 为 1 时，选择 CLKIN2。参考时钟在 CLKIN1 和 CLKIN2 之间动态切换，切换后需要对 PLL 复位。

对于 PPLL 输入时钟需要注意以下几点：

- 当输入时钟丢失或者不稳定时，需要输入时钟恢复稳定后对 PPLL 复位
- 上电以及上电后无输入时钟，PPLL 会输出一个低频时钟信号，此时 LOCK 信号为低电平，如果想让输出时钟为 0，可以使用关电模式或者时钟输出门控，详见 2.9.6.3 和 2.9.6.10 小节

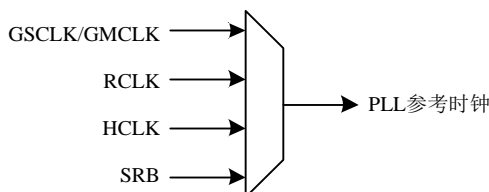


图 2-57 GTP_PPLL 参考时钟源

2.9.6.2 PPLL 复位

RST 复位信号释放之后，PLL 开始进入锁定状态，经过 T_{LOCK} 时间之后完成 PLL 频率锁定。LOCK 信号从低电平到变成高电平之前 CLKOUT 会从低电平逐渐开始翻转，此时 CLKOUT 是不稳定的，LOCK 信号变成高电平后 CLKOUT 才保持稳定。RST 复位信号的时序如下图所示，高电平有效。

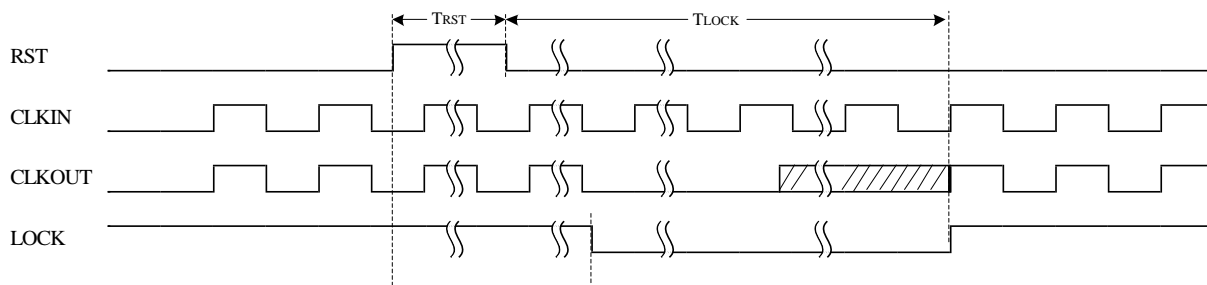


图 2-58 GTP_PPLL 复位时序

T_{RST} 和 T_{LOCK} 性能指标见《DS04001_Logos2 系列 FPGA 器件数据手册》。

2.9.6.3 PPLL 关电

当 PLL 不使用时，可通过将 PLL_PWD 信号置为高电平关闭 PLL 电源，以便节省功耗。如果要再次使用 PLL，将 PLL_PWD 信号置为低电平，经过 T_{LOCK} 时间后 PLL 重新锁定。LOCK 信号从低电平到变成高电平之前 CLKOUT 会从低电平逐渐开始翻转，此时 CLKOUT 是不稳定的，LOCK 信号变成高电平后 CLKOUT 才保持稳定。

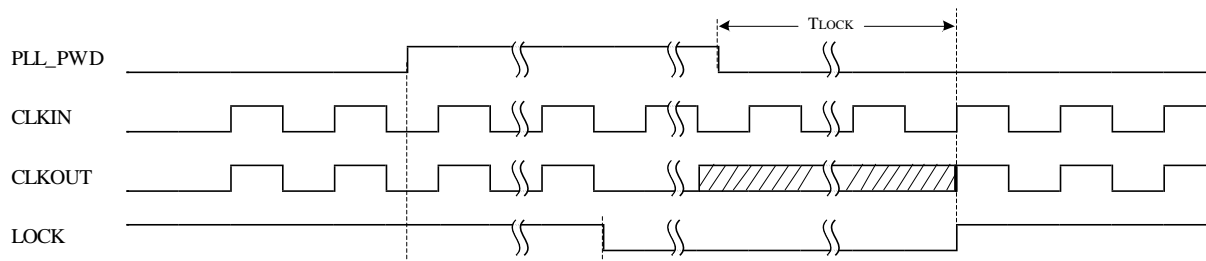


图 2-59 GTP_PPLL 关电时序

2.9.6.4 反馈时钟选择

PPLL 的反馈时钟输入源可以分为 PPLL 内部反馈和 PPLL 外部反馈两种，其中反馈源支持 CLKOUT[0]~ CLKOUT[4],CLKOUTF。

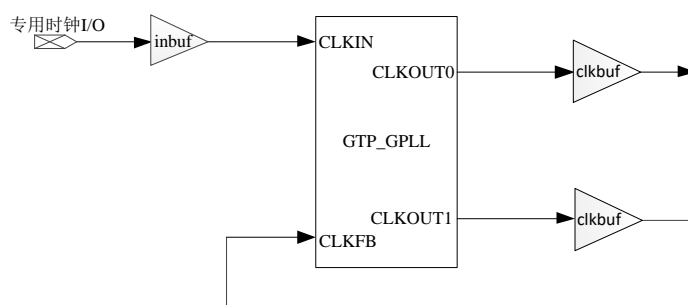


图 2-60 GTP_PPLL 外部反馈示意图

如图所示，选择外反馈时，反馈时钟通过全局时钟或者水平时钟 buffer 走专用时钟路径到达反馈时钟输入，而内反馈模式下，反馈时钟输出在 PLL 内部直接到达反馈时钟输入，相对外反馈受干扰小。

表 2-33 反馈时钟选择列表

INTERNAL_FB	EXTERNAL_FB	反馈模式
“CLKOUT0”	“DISABLE”	内部反馈模式，选择输出 divider0；
“CLKOUT1”	“DISABLE”	内部反馈模式，选择输出 divider1；

INTERNAL_FB	EXTERNAL_FB	反馈模式
“CLKOUT2”	“DISABLE”	内部反馈模式，选择输出 divider2;
“CLKOUT3”	“DISABLE”	内部反馈模式，选择输出 divider3;
“CLKOUT4”	“DISABLE”	内部反馈模式，选择输出 divider4;
“CLKOUTF”	“DISABLE”	内部反馈模式，选择反馈 F divider;
“DISABLE”	“CLKOUT0”	外部反馈模式，选择输出 divider0;
“DISABLE”	“CLKOUT1”	外部反馈模式，选择输出 divider1;
“DISABLE”	“CLKOUT2”	外部反馈模式，选择输出 divider2;
“DISABLE”	“CLKOUT3”	外部反馈模式，选择输出 divider3;
“DISABLE”	“CLKOUT4”	外部反馈模式，选择输出 divider4;
“DISABLE”	“CLKOUTF”	外部反馈模式，选择反馈 F divider;

2.9.6.5 输出频率编程

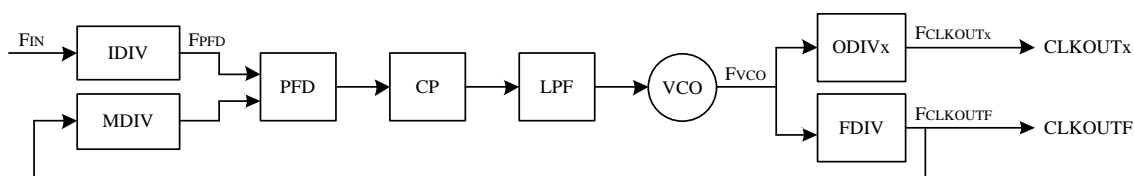


图 2-61 GTP_PPLL 输出频率编程示意图

参考时钟（Fin）经过可编程输入分频器（IDIV）得到鉴频鉴相器（PFD）的一个参考时钟，反馈时钟经过可编程反馈分频器（FDIV）和（MDIV）得到 PFD 另一个参考时钟，PFD 比较这两个时钟的相位和频率，然后产生一个信号驱动电荷泵（CP），CP 产生的电流信号经过环路滤波器（LPF），LPF 产生一个参考电压给压控振荡器（VCO）；PFD 给 CP 和 LPF 提供相位超前或滞后的信号来不断调整 VCO 工作的频率，以此调节频率和相位。例如当 VCO 经过反馈分频器后仍高于参考频率，PFD 产生一个向下调节的信号，使得 CP 和 LPF 减小电压控制 VCO 降低输出频率，当 VCO 经过反馈分频器后仍低于参考频率，PFD 产生一个向上调节的信号，使得 CP 和 LPF 提高电压控制 VCO 增加输出频率。VCO 产生 8 相位输出时钟，通过 8 相位时钟可实现相移粗调和细调，每一个输出相移时钟都可以作为 FDIV 的输入，共有 7 个输出分频器，6 个 ODIV 和一个 FDIV，每个分频器都是独立的，VCO 输出的时钟经过这些分频器得到最终的输出时钟频率。

表 2-34 PPLL 输出频率编程参数说明

功能名称	描述
FIN	输入时钟频率
FPFD	PFD 输入参考时钟频率
FVCO	VCO 输出时钟频率

功能名称	描述
F _{CLKOUTx}	CLKOUT0~4/CLKOUTPHY 输出时钟频率
F _{CLKOUTF}	CLKOUTF 输出时钟频率
IDIV	输入分频器分频比
MDIV	反馈分频器 M 分频比
ODIV _x	输出分频器分频比，x 代表 0~4/PHY，STATIC_RATIO0~4 对应 CLKOUT0~4，STATIC_RATIOPHY 对应 CLKOUTPHY
FDIV	反馈分频器 F 分频比

注：IDIV 对应参数 STATIC_RATIOI，MDIV 对应参数 STATIC_RATIO_M，ODIV_{0~4} 对应参数 STATIC_RATIO0~4，ODIV_{phy} 对应参数 STATIC_RATIOPHY，FDIV 对应参数 STATIC_RATIOF

若内部反馈或外部反馈选择的反馈分频器分频比为 FBDIV，各时钟频率计算公式如下：

$$F_{PFD} = \frac{F_{IN}}{IDIV}$$

$$F_{VCO} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV}$$

$$F_{CLKOUTx} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV \times ODIV_x}$$

$$F_{CLKFBOUT} = \frac{F_{IN} \times MDIV \times FBDIV}{IDIV \times FDIV}$$

其中 FBDIV 为反馈环路选择的分频器分频比，通过配置 INTERNAL_FB 或者 EXTERNAL_FB 选择 FDIV 或 ODIV_x 中某个分频器。

2.9.6.6 环路带宽

PPLL 环路带宽可配置为低带宽（LOW）、中带宽（OPTIMIZED）和高带宽（HIGH），低带宽和中带宽配置参数一样。

考虑到实际应用中输入时钟非理想时钟，不同带宽配置对 PLL 影响不同：

- 若期望 PLL 输出时钟有较小 Jitter，则配置为低带宽或中带宽，此时锁定时间长
- 若期望 PLL 锁定时间短，则配置为高带宽，此时输出时钟 Jitter 变差

2.9.6.7 相位调整

相位调整包含相位粗调、相位细调两种方式，其中相位粗调和相位细调既可以通过静态配置也可以通过 APB 接口调整。

➤ 相位调整方式

• 相位粗调

通过配置参数 `STATIC_CPHASE0/1/2/3/4/F` 和 `STATIC_CPHASEPHY` 可以实现输出时钟 `CLKOUT0/1/2/3/4/F` 和 `CLKOUTPHY` 相位粗调。

相对于输入的 VCO 时钟, 输出时钟相位粗调的步阶是 T_{vco} , 调整范围是 $[0 \sim N_{div}-1] * T_{vco}$, 调整度数步阶是 $360^\circ / N_{div}$ 。

N_{div} 是输出分频比 $ODIV_x$ 和反馈分频比 $FDIV$ 。

• 相位细调

通过配置参数 `STATIC_PHASE0/1/2/3/4/F` 和 `STATIC_PHASEPHY` 可以实现输出时钟 `CLKOUT0/1/2/3/4/F` 和 `CLKOUTPHY` 相位细调。

相对于输入的 VCO 时钟, 输出时钟相位细调的步阶是 T_{vco} , 调整范围是 $(0 \sim 7/8) T_{vco}$, 调整度数步阶是 $45^\circ / N_{div}$, 调整范围是 $(0 \sim 45^\circ * 7) / N_{div}$ 。

注: $N_{div} \in \{STATIC_RATIO0 \sim 4, STATIC_RATIOPHY, STATIC_RATIOF\}$, 是输出时钟 `CLKOUT0/1/2/3/4/PHY/F` 的分频系数

➤ 相位调整模式

• 静态配置

输出时钟 `CLKOUT0/1/2/3/4/PHY/F` 相位细调静态配置分别由 `STATIC_PHASE0/1/2/3/4/PHY/F` 控制;

输出时钟 `CLKOUT0/1/2/3/4/PHY/F` 相位粗调静态配置分别由 `STATIC_CPHASE0/1/2/3/4/PHY/F` 控制;

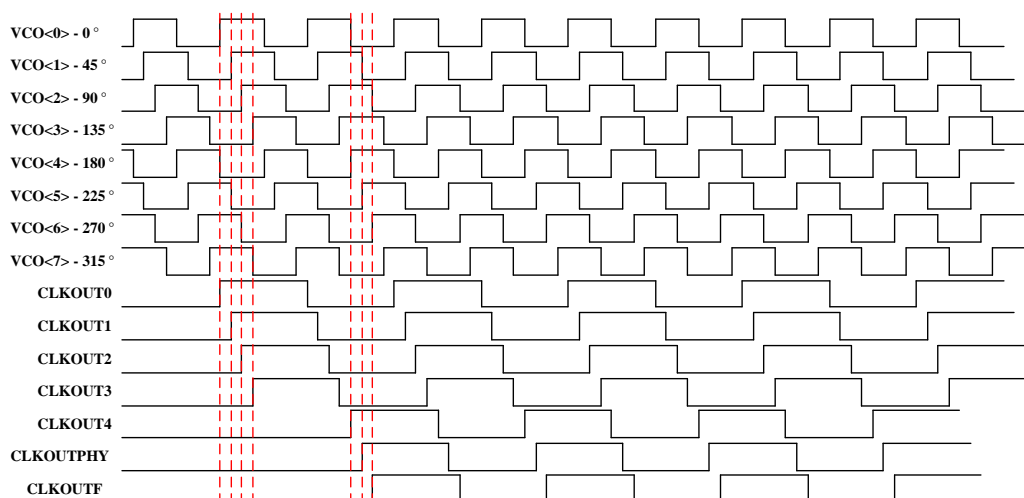


图 2-62 GTP_PPLL 相位调整_静态配置

如下表所示, `CLKOUT0` 由 `VCO<0>` 二分频产生, 细调相位为 0, 粗调相位为 0, 以

CLKOUT0 为基准，各个输出时钟相位关系如下：

根据公式，由于 $ODIV0=2$ ，则细调相位最小 $step=45^\circ/2=22.5^\circ$ ；粗调相位最小 $step=180^\circ$ 。

表 2-35 输出时钟相位关系

输出时钟	细调配置	粗调配置	相位关系
CLKOUT0	0	0	0
CLKOUT1	$1 \cdot T_{vco}/8$	0	22.5°
CLKOUT2	$2 \cdot T_{vco}/8$	0	45°
CLKOUT3	$3 \cdot T_{vco}/8$	0	67.5°
CLKOUT4	$4 \cdot T_{vco}/8$	T_{vco}	270°
CLKOUTPHY	$5 \cdot T_{vco}/8$	T_{vco}	292.5°
CLKOUTF	$6 \cdot T_{vco}/8$	T_{vco}	315°

2.9.6.8 占空比可编程

输出 divider 支持可编程的 Duty Cycle 最小 step 为 $50\%/odiv$ ，odiv 为 divider 分频系数。STATIC_DUTY0/1/2/3/4/PHY/F 决定了分频后 duty cycle，表达式如下，duty 为 duty cycle 配置值， $duty\ cycle = (50\%/odiv) \cdot duty$ 。

注意：

- 1.DUTY 的设置值受到 odiv 值（输出分频）限制： $odiv>1$ 时， $2 \leq duty \leq 2 \cdot odiv - 1$ ； $odiv=1$ ，duty 无效，不起作用，默认 50% 输出。
- 2.分数分频不支持占空比编程。

2.9.6.9 扩频时钟输入

支持频谱扩展时钟输入，调制频率小于 PPLL 带宽。

2.9.6.10 时钟输出门控

GTP_PPLL 输入信号 CLKOUT0_SYN 用来使能和关闭 CLKOUT0 输出时钟，当用户不需要 CLKOUT0 时钟时，可以关闭该时钟以节省 Clock Tree 和 Fabric 功耗。

CLKOUT0_SYN 信号可选，当用户配置 CLKOUT0_SYN_EN 为“TRUE”时，该信号有效。

同样地，输入信号 CLKOUT1~4/PHY/F_SYN 分别与配置位 CLKOUT1~4/PHY/F_SYN_EN

配合，使能和关闭 CLKOUT1~4/PHY/F 时钟。

以输出时钟 CLKOUT0 为例，CLKOUT0_SYN 电平由 0 跳变到 1 时，经过 3 个 CLKIN 下降沿 CLKOUT0 输出低电平，CLKOUT0_SYN 电平由 1 跳变到 0 时，经过 3 个 CLKIN 下降沿 CLKOUT0 输出有效。功能示意如下图所示：

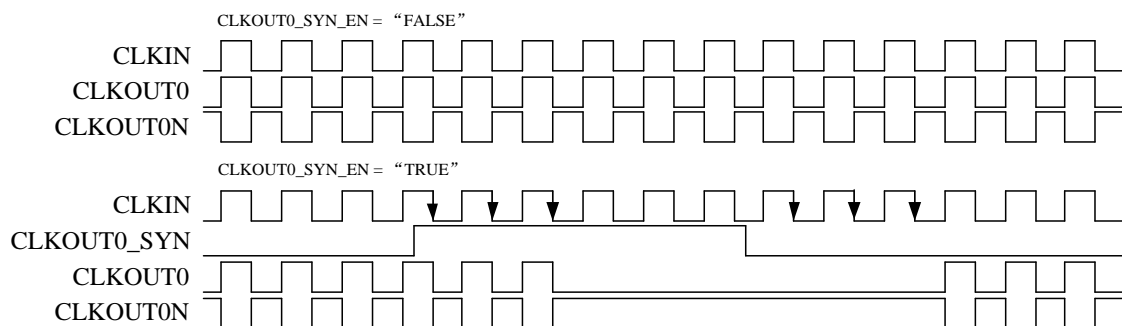


图 2-63 GTP_PPLL 输出时钟 GATE 时序图

2.9.6.11 LOCK 指示

PPLL 支持两种 LOCK 模式，由配置参数 LOCK_MODE 进行选择，PLL 工作时，当 LOCK 拉低时需要复位 PLL。

表 2-36 PPLL LOCK 指示模式

参数名	参数值	功能描述
LOCK_MODE	0	实时监测 PLL LOCK 的状态，一旦进入非正常状态则会拉低 LOCK.
	1	LOCK 不会实时监控（指示）PLL 工作状态；LOCK 一旦拉高则不会拉低，除非复位或 Power Down.

2.9.7 GTP_PPLL 应用模式

2.9.7.1 内部反馈模式

内部反馈模式主要针对某些应用场景下，对 PPLL 输出时钟具有高性能的要求，比如频率综合、低抖动时钟输出等。

内部反馈模式下，反馈时钟直接连接到 PPLL 的输入端，不经过任何其他的电路，这样就免除了其他电路引入的噪声，拥有更好的抖动性能。

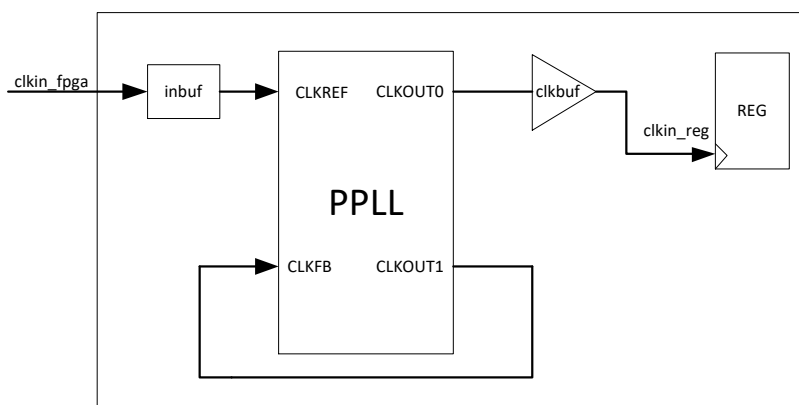


图 2-64 PPLL 内部反馈模式示意图

2.9.7.2 源同步模式

采用外部反馈实现源同步模式，源同步模式主要针对 PPLL 输出时钟驱动内部寄存器时，需要保证输入时钟与数据保持一定的相位关系，如下图所示。PPLL 锁定时，b/e 两点相位是对齐的，源同步模式下，PPLL 输出时钟 clkout0 送给寄存器的路径与反馈时钟 clkout1 所经历的路径，必须对称平衡，且使用同一 regional buffer，也就是保证 c 点与 e 点时钟相位对齐，即 b 点与 c 点时钟相位对齐；同时，可以通过 PPLL 相位调整功能补偿数据管脚到寄存器数据端输入和时钟管脚到 PPLL 输入参考时钟之间的 skew(偏移)，即 d 点到 f 点和 a 点到 b 点的偏移，这样可以在 FPGA 内部 REG 处，可以实现数据与时钟之间相位边沿或中心对齐，具体是 0° 还是 90° 对齐，取决于用户的要求，具体的相移要求可以通过输出时钟的分频器的 phase shift 功能予以实现。

例如，输入管脚时钟上升沿处于数据的中央进入 FPGA，可以采用 PPLL 输出 phase shift 功能，移动 clkout0 的相位，在 FPGA 内部的寄存器，可以实现时钟上升沿对准数据的中央。此时，如果输入时钟 clk_in_fpga 进入 PPLL 的参考输入与 data 进入 register 之间如没有任何 skew，比如走专用时钟路径，即 skew 很小，此时，PPLL 可以不用任何 phase shift 调整，自动实现上述的时钟上升沿与数据中央中心对齐。

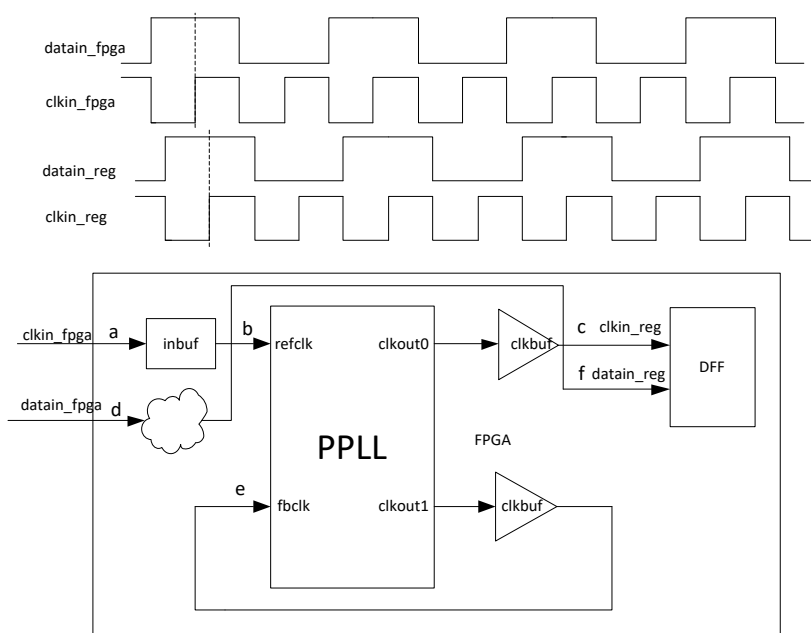


图 2-65 PPLL 源同步模式示意图

2.9.7.3 零延迟模式

采用外部管脚反馈实现零延迟模式，主要针对只有一个输入参考时钟情况下，且 FPGA 驱动多个外部器件情形时，需要保证外部器件的输入时钟与 FPGA 输入管脚的时钟对齐，如下图所示。

零延迟模式下，输入时钟与反馈时钟需要使用相同的 IO 标准，以保证 a 点与 d 点时钟相位对齐，b 点与 e 点相位对齐。同时，板上反馈时钟 trace 与驱动外部器件的时钟 trace 相互匹配等长，则 d 点与 c 点时钟相位对齐，于是 a、c 点时钟对齐，实现零延迟缓冲的功能。

实际情况下，由于外部器件的负载寄生电容与 FPGA 输入管脚的寄生电容不一致，会导致 c 点与 d 点的信号上升沿或下降沿时间不一致，会导致延迟略有差异。因此这个 a/c 时钟 skew 的差异，客户需要在时序收敛时必须予以考虑，留有一定的余量。

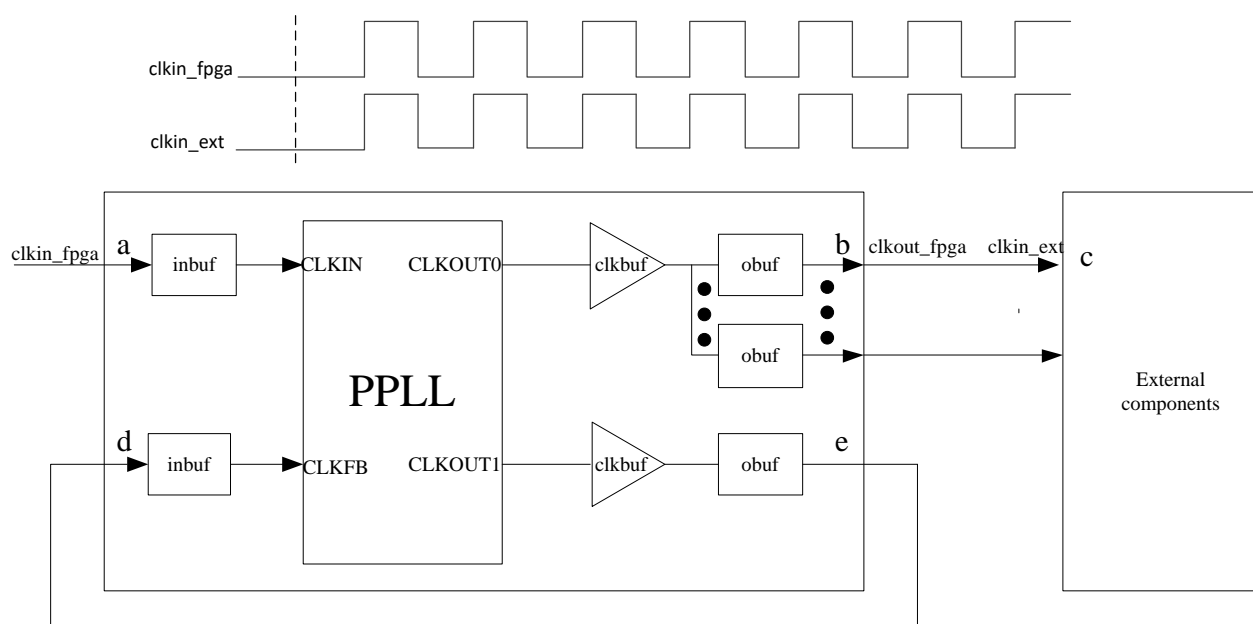


图 2-66 PPLL 零延迟模式示意图

免责声明

版权声明

本文档版权归深圳市紫光同创电子有限公司所有，并保留一切权利。未经书面许可，任何公司和个人不得将此文档中的任何部分公开、转载或以其他方式披露、散发给第三方。否则，公司必将追究其法律责任。

免责声明

- 1、本文档仅提供阶段性信息，所含内容可根据产品的实际情况随时更新，恕不另行通知。如因本文档使用不当造成的直接或间接损失，本公司不承担任何法律责任。
- 2、本文档按现状提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。
- 3、公司保留任何时候在不事先声明的情况下对公司系列产品相关文档的修改权利。