

## 引言

LatticeECP2™ 和 LatticeECP2M™ sysIO™ 缓冲器让设计人员能够方便地与使用先进的系统 I/O 标准的其他器件接口。本技术说明阐述了现行的 sysIO 标准以及如何使用莱迪思的 ispLEVER® 设计软件来进行实现。

## sysIO 缓冲器概述

LatticeECP2/M sysIO 接口包含多个可编程 I/O 单元 (PIC) 块。每个 PIC 块包含两个可编程 I/O (PIO)，PIOA 和 PIOB，分别连接到各自的 sysIO 缓冲器。可以将两个相邻的 PIO 用作差分 I/O 对 (标为“T”和“C”)。

每个可编程 I/O (PIO) 包括一个 sysIO 缓冲器和 I/O 逻辑 (IOLOGIC)。LatticeECP2/M sysIO 缓冲器支持各种单端和差分信号标准。sysIO 缓冲器还支持与 DDR 存储器接口所需的 DQS 选通脉冲信号。LatticeECP2/M 中每 16/18 个 PIO 中有一个包含了一个用以产生 DQS 信号的延迟单元。总线上的 DQS 信号是用来选通从存储器到输入寄存器块的 DDR 数据。欲了解更多关于 sysIO 缓冲器架构的信息，请参阅 [LatticeECP2/M 系列数据手册](#)。

IOLOGIC 包括输入、输出和三态寄存器，与必要的时钟和数据选择逻辑一起实现单倍数据速率 (SDR) 和双倍数据速率 (DDR) 应用。IOLOGIC 内的可编程延时线和专用逻辑是用来提供输入时钟和数据信号相位移位以及 DDR 存储器中 DQS 输入所需的延时。有关 IOLOGIC 中的 DDR 实现和 DDR 存储器接口支持的详细讨论请参见 TN1105，[LatticeECP2/M 高速 I/O 接口](#)。

## sysIO 标准支持

LatticeECP2/M sysIO 缓冲器支持单端和差分标准。单端标准可进一步细分为内部比例电平标准 (internally ratioed standard)，例如 LVCMOS、LVTTL 和 PCI，以及外部参考电平标准 (externally referenced standard)，例如 HSTL 和 SSTL。缓冲器支持 LVTTL、LVCMOS 1.2, 1.5, 1.8, 2.5 和 3.3V 标准。在 LVCMOS 和 LVTTL 模式下，缓冲器有用于驱动电流强度、总线维持 (弱上拉、弱下拉或总线保持锁存) 的独立配置选项。其他支持的单端标准还包括 SSTL 和 HSTL。支持的差分标准包括 LVDS、RSDS、BLVDS、LVPECL、差分 SSTL 和差分 HSTL。表 1 和表 2 列出了 LatticeECP2/M 器件支持的 sysIO 标准。

**表 1. 支持的输入标准**

| 输入标准                  | V <sub>REF</sub> (标称值) | V <sub>CCIO</sub> <sup>1</sup> (标称值) |
|-----------------------|------------------------|--------------------------------------|
| <b>单端接口</b>           |                        |                                      |
| LVTTL                 | —                      | —                                    |
| LVCMOS33              | —                      | —                                    |
| LVCMOS25              | —                      | —                                    |
| LVCMOS18              | —                      | 1.8                                  |
| LVCMOS15              | —                      | 1.5                                  |
| LVCMOS12              | —                      | —                                    |
| PCI 33                | —                      | 3.3                                  |
| HSTL18 Class I, II    | 0.9                    | —                                    |
| HSTL15 Class I        | 0.75                   | —                                    |
| SSTL3 Class I, II     | 1.5                    | —                                    |
| SSTL2 Class I, II     | 1.25                   | —                                    |
| SSTL18 Class I, II    | 0.9                    | —                                    |
| <b>差分接口</b>           |                        |                                      |
| 差分 SSTL18 Class I, II | —                      | —                                    |
| 差分 SSTL2 Class I, II  | —                      | —                                    |

Lattice Semiconductor Corp. 2009 版权所有 © 所有莱迪思的商标、注册商标、图案和标识符均在 [www.latticesemi.com/legal](http://www.latticesemi.com/legal) 网站上列出。所有其它品牌或产品名称均为其所有者的商标或注册商标。此处的参数规格和信息可能会更改，恕不另行通知。中文翻译文档仅为为您提供方便。莱迪思将尽力为您提供准确的中文翻译文档，但鉴于翻译的难度，译文可能会与英文文档存在一些微小差别，其准确性也难以保证。请参考英文源文件，获取最新、最准确的信息。所有的翻译文档中的信息均以英文源文件为准。

表 1. 支持的输入标准 (续)

| 输入标准                             | V <sub>REF</sub> (标称值) | V <sub>CCIO</sub> <sup>1</sup> (标称值) |
|----------------------------------|------------------------|--------------------------------------|
| 差分 SSTL3 Class I, II             | —                      | —                                    |
| 差分 HSTL15 Class I                | —                      | —                                    |
| 差分 HSTL18 Class I, II            | —                      | —                                    |
| LVDS, MLVDS, LVPECL, BLVDS, RSDS | —                      | —                                    |

1 如未指明, V<sub>CCIO</sub> 可设为有效工作范围内的任意值。

表 2. 支持的输出标准

| 输出标准                  | 驱动                         | V <sub>CCIO</sub> (标称值) |
|-----------------------|----------------------------|-------------------------|
| <b>单端接口</b>           |                            |                         |
| LVTTL                 | 4mA, 8mA, 12mA, 16mA, 20mA | 3.3                     |
| LVC MOS33             | 4mA, 8mA, 12mA 16mA, 20mA  | 3.3                     |
| LVC MOS25             | 4mA, 8mA, 12mA, 16mA, 20mA | 2.5                     |
| LVC MOS18             | 4mA, 8mA, 12mA, 16mA       | 1.8                     |
| LVC MOS15             | 4mA, 8mA                   | 1.5                     |
| LVC MOS12             | 2mA, 6mA                   | 1.2                     |
| LVC MOS33, 开漏         | 4mA, 8mA, 12mA 16mA, 20mA  | —                       |
| LVC MOS25, 开漏         | 4mA, 8mA, 12mA 16mA, 20mA  | —                       |
| LVC MOS18, 开漏         | 4mA, 8mA, 12mA 16mA        | —                       |
| LVC MOS15, 开漏         | 4mA, 8mA                   | —                       |
| LVC MOS12, 开漏         | 2mA, 6mA                   | —                       |
| PCI33/PCIX            | N/A                        | 3.3                     |
| HSTL18 Class I        | 8mA, 12mA                  | 1.8                     |
| HSTL18 Class II       | N/A                        | 1.8                     |
| HSTL15 Class I        | 4mA, 8mA                   | 1.5                     |
| SSTL3 Class I, II     | N/A                        | 3.3                     |
| SSTL2 Class I         | 8mA, 12mA                  | 2.5                     |
| SSTL2 Class II        | 16mA, 20mA                 | 2.5                     |
| SSTL18 Class I        | N/A                        | 1.8                     |
| SSTL18 Class II       | 8mA, 12mA                  | 1.8                     |
| <b>差分接口</b>           |                            |                         |
| 差分 SSTL3, Class I, II | N/A                        | 3.3                     |
| 差分 SSTL2, Class I     | 8mA, 12mA                  | 2.5                     |
| 差分 SSTL2, Class II    | 16mA, 20mA                 | 2.5                     |
| 差分 SSTL18, Class I    | N/A                        | 1.8                     |
| 差分 SSTL18, Class II   | 8mA, 12mA                  | 1.8                     |
| 差分 HSTL18, Class I    | 8mA, 12mA                  | 1.8                     |
| 差分 HSTL18, Class II   | N/A                        | 1.8                     |
| 差分 HSTL15, Class I    | 4mA, 8mA                   | 1.5                     |
| LVDS                  | N/A                        | 2.5                     |
| MLVDS <sup>1</sup>    | N/A                        | 2.5                     |
| BLVDS <sup>1</sup>    | N/A                        | 2.5                     |
| LVPECL <sup>1</sup>   | N/A                        | 3.3                     |
| RSDS <sup>1</sup>     | N/A                        | 2.5                     |

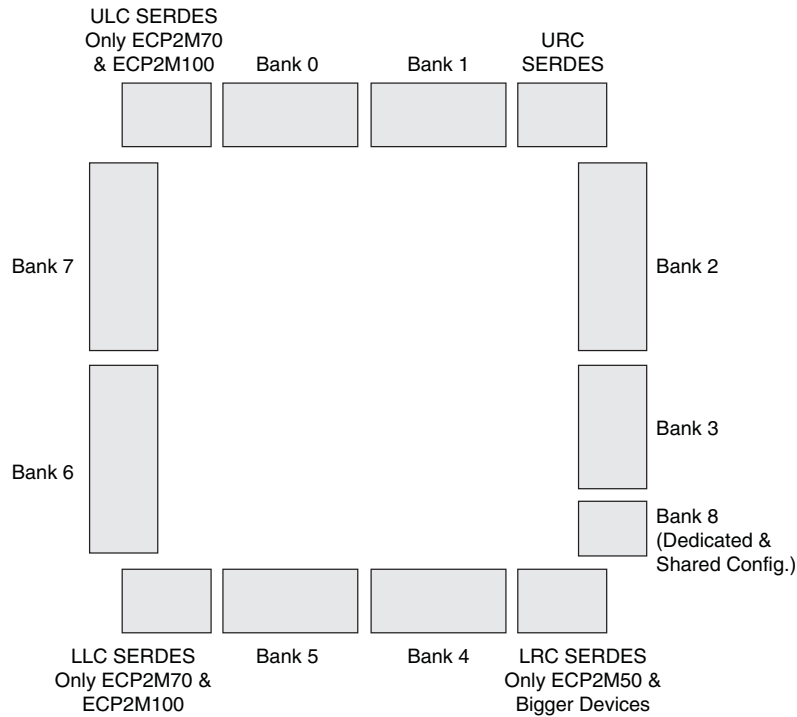
1. 使用外部电阻仿真。

## sysIO Bank 机制

LatticeECP2/M 器件有 8 个通用可编程 sysIO bank 和第九个配置 bank。8 个通用 sysIO bank 中的每一个都有一个 VCCIO 供电电压和两个参考电压，VREF1 和 VREF2。图 1 展示了 8 个通用 bank 和配置 bank 及其相关的电源。Bank 8 是专用于配置逻辑的 bank，并有 7 个专用配置 I/O 和 14 个多路复用配置 I/O。Bank 8 有电源焊盘（VCCIO 和 VCCAUX），但没有独立的 VREF 焊盘。Bank 8 中的 I/O 连接到 Bank 3 的 VREF。

在器件的上侧和下侧的 bank 中，sysIO 缓冲器对包括 2 个单端输出驱动器和 2 组单端输入缓冲器（比例电平和参考电平）。左侧和右侧的 sysIO 缓冲器对包括 2 个单端输出驱动器和 2 组单端输入缓冲器（比例电平和参考电平）。参考输入缓冲器也可以配置为差分输入。在 50% 的缓冲器对中有一个差分输出驱动器。两个配对的焊盘被称为“true”和“comp”，“true”连到差分输入缓冲器的正极，“comp”连到差分输入缓冲器的负极。

图 1. LatticeECP2M sysIO Bank



Note: URC = upper right corner, LRC = lower right corner, ULC = upper left corner and LLC = lower left corner.

### V<sub>CCIO</sub> (1.2V/1.5V/1.8V/2.5V/3.3V)

一共有 8 个 V<sub>CCIO</sub> 电源, V<sub>CCIO0</sub> - V<sub>CCIO7</sub>。每个 bank 有一个独立的 V<sub>CCIO</sub> 电源为单端输出驱动器和比例电平输入缓冲器如 LVTTTL、LVC MOS 和 PCI 供电。LVTTTL、LVC MOS3.3、LVC MOS2.5 和 LVC MOS1.2 还有固定阈值选项, 可用于任意 bank。加到 bank 上的 V<sub>CCIO</sub> 电压决定了该 bank 所支持的比例电平输入标准。V<sub>CCIO</sub> 电压还被用于对差分输出驱动器进行供电。此外, V<sub>CCIO8</sub> 用于对 sysCONFIG™ 信号进行供电。

### V<sub>CCAUX</sub> (3.3V)

除了 bank 上的 V<sub>CCIO</sub> 电源, 器件还有一个 V<sub>CC</sub> 内核逻辑电源和 1 个 V<sub>CCAUX</sub> 辅助电源, V<sub>CCAUX</sub> 为差分 and 参考输入缓冲器供电。V<sub>CCAUX</sub> 用于提供 I/O 参考电压所需要的 3.3V, 以满足驱动器和输入缓冲器的共模工作电压范围。

### V<sub>CCJ</sub> (1.2V/1.5V/1.8V/2.5V/3.3V)

JTAG 引脚有一个独立于 bank 内 V<sub>CCIO</sub> 电源的 V<sub>CCJ</sub> 电源。V<sub>CCJ</sub> 决定了 LVC MOS JTAG 引脚的电气特性, 包括输出高电平和输入阈值。

表 3 为所有所需电源的总结列表。

表 3. 电源

| 电源                 | 说明               | 值 <sup>1</sup>           |
|--------------------|------------------|--------------------------|
| V <sub>CC</sub>    | 核电源              | 1.2V                     |
| V <sub>CCIO</sub>  | I/O 和配置 bank 的电源 | 1.2V/1.5V/1.8V/2.5V/3.3V |
| V <sub>CCAUX</sub> | 辅助电源             | 3.3V                     |
| V <sub>CCJ</sub>   | JTAG 引脚的电源       | 1.2V/1.5V/1.8V/2.5V/3.3V |

1. 关于建议使用的最小和最大值, 请参见 [LatticeECP2/M 系列数据手册](#)。

### 输入参考电压 (V<sub>REF1</sub>, V<sub>REF2</sub>)

每个 bank 最多可支持两个独立的 V<sub>REF</sub> 输入电压, V<sub>REF1</sub> 和 V<sub>REF2</sub>, 用于设置参考输入缓冲器的阈值。这些 V<sub>REF</sub> 引脚在 bank 内的位置是预先确定的。如果 bank 不需要 V<sub>REF</sub> 电压, 这些引脚还可以用作普通的 I/O。

### V<sub>REF1</sub> 用于 DDR 存储器接口

当连接到 DDR 存储器时, V<sub>REF1</sub> 输入必须被用作来自存储器的 DQS 和 DQ 输入的参考电压。V<sub>REF1</sub> 和地之间使用一个分压器来产生一个片上参考电压, 用于 DQS 变化检测电路。这个分压器仅用于 V<sub>REF1</sub>, 而不能用于 V<sub>REF2</sub>。有关 DQS 变化检测逻辑及其实现的进一步信息, 请参见 TN1105, [LatticeECP2/M 高速 I/O 接口](#)。DDR1 符合 SSTL25\_II 信号标准, DDR2 符合 SSTL18\_II 信号标准。

### 一个 Bank 内的混合电压支持

LatticeECP2/M sysIO 缓冲器连到三个并行的比例电平输入缓冲器。这三个并行的缓冲器连接到 V<sub>CCIO</sub>、V<sub>CCAUX</sub> 和 V<sub>CC</sub>, 支持追踪 V<sub>CCIO</sub> 的阈值以及 3.3V (V<sub>CCAUX</sub>) 和 1.2V (V<sub>CC</sub>) 输入的固定阈值。这使得比例电平缓冲器的输入阈值可按每个引脚来设置, 而不只是追踪 V<sub>CCIO</sub> 电压。该选项可用于所有 1.2V、2.5V 和 3.3V 比例电平输入并且独立于 bank 的 V<sub>CCIO</sub> 电压。例如, 如果 bank 的 V<sub>CCIO</sub> 为 1.8V, 可以实现固定阈值的 1.2V 和 3.3V 比例电平输入缓冲器, 以及追踪阈值的 2.5V 比例电平输入。

在器件配置之前, 比例电平输入阈值总是追踪 bank 的 V<sub>CCIO</sub>。这个选项只需要配置即可生效。bank 内的输出标准总是由 V<sub>CCIO</sub> 设置。表 4 显示了可以在同一个 bank 内混合的 sysIO 标准。

表 4. 混合电压支持

| V <sub>CCIO</sub> | 输入 sysIO 标准 |      |      |      |      | 输出 sysIO 标准 |      |      |      |      |
|-------------------|-------------|------|------|------|------|-------------|------|------|------|------|
|                   | 1.2V        | 1.5V | 1.8V | 2.5V | 3.3V | 1.2V        | 1.5V | 1.8V | 2.5V | 3.3V |
| 1.2V              | 是           |      |      | 是    | 是    | 是           |      |      |      |      |
| 1.5V              | 是           | 是    |      | 是    | 是    |             | 是    |      |      |      |
| 1.8V              | 是           |      | 是    | 是    | 是    |             |      | 是    |      |      |
| 2.5V              | 是           |      |      | 是    | 是    |             |      |      | 是    |      |
| 3.3V              | 是           |      |      | 是    | 是    |             |      |      |      | 是    |

每个 Bank 支持的 sysIO 标准

表 5. 每个 Bank 支持的 I/O 标准

| 说明         | 上侧 Bank 0-1  | 右侧 Bank 2-3  | 下侧 Bank 4-5  | 左侧 Bank 6-7   |
|------------|--|--|--|---|
| I/O 缓冲器的类型 | 单端   | 单端和差分  | 单端   | 单端和差分   |
| 支持的输出标准    | LVTTL<br>LVCMOS33<br>LVCMOS25<br>LVCMOS18<br>LVCMOS15<br>LVCMOS12<br><br>SSTL18 Class I, II<br>SSTL25 Class I, II<br>SSTL33 Class I, II<br><br>HSTL15 Class I<br>HSTL18_I, II<br><br>SSTL18D Class I, II<br>SSTL25D Class I, II<br>SSTL33D Class I, II<br><br>HSTL15D Class I<br>HSTL18D Class I, II<br><br>LVDS25E <sup>1</sup><br>LVPECL <sup>1</sup><br>BLVDS <sup>1</sup><br>RSDS <sup>1</sup> | LVTTL<br>LVCMOS33<br>LVCMOS25<br>LVCMOS18<br>LVCMOS15<br>LVCMOS12<br><br>SSTL18 Class I, II<br>SSTL25 Class I, II<br>SSTL33 Class I, II<br><br>HSTL15 Class I<br>HSTL18 Class I, II<br><br>SSTL18D Class I, II<br>SSTL25D Class I, II<br>SSTL33D Class I, II<br><br>HSTL15D Class I<br>HSTL18D Class I, II<br><br>LVDS<br>LVDS25E <sup>1</sup><br>LVPECL <sup>1</sup><br>BLVDS <sup>1</sup><br>RSDS <sup>1</sup> | LVTTL<br>LVCMOS33<br>LVCMOS25<br>LVCMOS18<br>LVCMOS15<br>LVCMOS12<br><br>SSTL18 Class I, II<br>SSTL2 Class I, II<br>SSTL3 Class I, II<br><br>HSTL15 Class I<br>HSTL18 Class I, II<br><br>SSTL18D Class I, II<br>SSTL25D Class I, II,<br>SSTL33D Class I, II<br><br>HSTL15D Class I<br>HSTL18D Class I, II<br><br>PCI33<br>LVDS25E <sup>1</sup><br>LVPECL <sup>1</sup><br>BLVDS <sup>1</sup><br>RSDS <sup>1</sup> | LVTTL<br>LVCMOS33<br>LVCMOS25<br>LVCMOS18<br>LVCMOS15<br>LVCMOS12<br><br>SSTL18 Class I, II<br>SSTL2 Class I, II<br>SSTL3 Class I, II<br><br>HSTL15 Class I, III<br>HSTL18 Class I, II, III<br><br>SSTL18D Class I, II<br>SSTL25D Class I, II,<br>SSTL33D_I, II<br><br>HSTL15D Class I<br>HSTL18D Class I, II<br><br>LVDS<br>LVDS25E <sup>1</sup><br>LVPECL <sup>1</sup><br>BLVDS <sup>1</sup><br>RSDS <sup>1</sup> |
| 输入         | 所有的单端, 差分  | 所有的单端, 差分  | 所有的单端, 差分  | 所有的单端, 差分   |
| 时钟输入       | 所有的单端, 差分  | 所有的单端, 差分  | 所有的单端, 差分  | 所有的单端, 差分   |
| PCI 支持     | PCI33 不带钳位二极管  | PCI33 不带钳位二极管  | PCI33 带有钳位二极管  | PCI33 不带钳位二极管<br>PCI33 带有钳位二极管 (ECP2M)  |
| LVDS 输出缓冲器 |  | LVDS (3.5mA) 缓冲器 <sup>2</sup>  |  | LVDS (3.5mA) 缓冲器 <sup>2</sup>   |

1. 这些差分标准通过使用带有外部电阻组合的互补 LVCMOS 驱动器来实现。
2. 仅 bank 内 50% 的 I/O 可用。

## LVC MOS 缓冲器配置

所有 LVC MOS 缓冲器的可编程上拉、可编程驱动强度和可编程摆率配置都可以在软件中进行设置。

### 总线维持电路

每个焊盘都有弱上拉、弱下拉和弱总线保持功能。上拉和下拉设置提供了固定的特性，有助于实现线逻辑，如：线或逻辑。但是，根据信号状态，电流要比其他选项下的稍大。总线保持选项锁存了最后一次驱动状态下的信号值，在最小的功耗的状态下以有效电平保持该信号。您还可以选择关闭总线保持电路，使功耗和输入漏电流最小化。请注意，在这种情况下，确保输入驱动为已知状态以避免输入缓冲器的不必要的功耗是很重要的。

### 可编程驱动强度

每个 LVC MOS 或 LV TTL 以及一些参考（SSTL 和 HSTL）输出缓冲器都具有一个可编程驱动强度选项。该选项可针对每个 I/O 单独配置。驱动强度可设置为 2mA、4mA、6mA、8mA、12mA、16mA 和 20mA。实际选项根据不同的 I/O 电压而不同。用户在选择驱动电流强度时，必须考虑每个 bank 最大允许的电流以及封装的热极限电流。表 6 显示了每个输出标准可选的驱动设置。

表 6. 单端缓冲器的可编程驱动值

| 单端 I/O 标准             | 可编程驱动电流 (mA)     |
|-----------------------|------------------|
| HSTL15_I/ HSTL15D_I   | 4, 8             |
| HSTL18_I/ HSTL18D_I   | 8, 12            |
| SSTL25_I/ SSTL25D_I   | 8, 12            |
| SSTL25_II/ SSTL25D_II | 16, 20           |
| SSTL18_II/SSTL18D_II  | 8, 12            |
| LVC MOS12             | 2, 6             |
| LVC MOS15             | 4, 8             |
| LVC MOS18             | 4, 8, 12, 16     |
| LVC MOS25             | 4, 8, 12, 16, 20 |
| LVC MOS33             | 4, 8, 12, 16, 20 |
| LV TTL                | 4, 8, 12, 16, 20 |

### 可编程摆率

每个 LVC MOS 或 LV TTL 输出缓冲引脚还有一个可编程输出摆率控制，可配置为低噪声或高速性能。每个 I/O 引脚有其单独的摆率控制。这使得设计者可以分别控制每个引脚的摆率。这个摆率控制影响上升和下降边沿。

### 开漏控制

所有的 LVC MOS 和 LV TTL 输出缓冲器可配置用作开漏输出。用户可通过在软件将 OPENDRAIN 属性设为 ON 来实现开漏输出。

### 差分 SSTL 和 HSTL 支持

单端驱动器的互补极“C”可以选择由驱动单端驱动器“true”极信号的互补信号来驱动。这使得一对单端驱动器能以信号间可能的最小相偏来驱动互补输出。这是用于驱动互补的 SSTL 和 HSTL 信号（这是同步 DRAM 和同步 SRAM 器件的差分 SSTL 和 HSTL 时钟输入分别所需的）。此功能还可以结合片外电阻来仿真 LVPECL 和 BLVDS 输出驱动器。

## 使用可编程 PCICLAMP 的 PCI 支持

每个 sysIO 缓冲区可以配置用于支持 PCI33。LatticeECP2 器件下侧的缓冲器或 LatticeECP2M 器件左侧和下侧的缓冲器，有一个可选的支持 PCI 的钳位二极管，可以选择在 ispLEVER 设计工具中进行设置。

可编程 PCICLAMP 可以打开或关闭。该选项可针对 LatticeECP2 器件下侧 bank 或 LatticeECP2M 器件左侧和下侧 bank 的每个 I/O 单独进行设置。

## 可编程输入延迟

每个输入在进入内核逻辑或输入寄存器之前，可以选择使其延迟。输入延迟主要用于当采用直接驱动主时钟时，实现输入寄存器的零保持时间。要实现零保持时间，数据的输入延迟至少要达到主时钟的注入延迟时间。此选项可以通过软件中的 FIXEDDELAY 属性针对每个 I/O 设为 ON 或 OFF。有关此属性的详细信息，请参见软件 sysIO 属性章节。附录 A 展示了此功能可以通过软件中使用 HDL 属性启用。

## 软件 sysIO 属性

sysIO 属性可以在 HDL 源代码中指定或使用 Preference Editor GUI 设置或直接在 ASCII preference 文件 (.prf) 中指定。本文档的附录 A、B 和 C 提供了如何使用上面所述的方法来进行属性设置的示例。本章节详细说明了这些属性。

## IO\_TYPE

这个属性是用来设置一个 I/O 的 sysIO 标准。属性名称中已经嵌入了设置这些 I/O 标准所需的 V<sub>CCIO</sub>。没有单独的属性来设置对 V<sub>CCIO</sub> 的要求。表 7 列出了可用的 I/O 类型。

表 7. IO\_TYPE 属性值

| sysIO 信号标准              | IO_TYPE               |
|-------------------------|-----------------------|
| 默认                      | LVC MOS25             |
| LVDS 2.5V               | LVDS25                |
| RSDS                    | RSDS                  |
| 仿真 LVDS 2.5V            | LVDS25E <sup>1</sup>  |
| 总线 LVDS 2.5V            | BLVDS25 <sup>1</sup>  |
| LVPECL 3.3V             | LVPECL33 <sup>1</sup> |
| HSTL18 Class I 和 II     | HSTL18_I, HTSL18_II   |
| 差分 HSTL 18 Class I 和 II | HSTL18D_I, HSTL18D_II |
| HSTL 15 Class I         | HSTL15_I              |
| 差分 HSTL 15 Class I      | HSTL15D_I             |
| SSTL 33 Class I 和 II    | SSTL33_I, SSTL33_II   |
| 差分 SSTL 33 Class I 和 II | SSTL33D_I, SSTL33D_II |
| SSTL 25 Class I 和 II    | SSTL25_I, SSTL25_II   |
| 差分 SSTL 25 Class I 和 II | SSTL25D_I, SSTL25D_II |
| SSTL 18 Class I 和 II    | SSTL18_I, SSTL18_II   |
| 差分 SSTL 18 Class I      | SSTL18D_I, SSTL18D_II |
| LVTTL                   | LVTTL33               |
| 3.3V LVC MOS            | LVC MOS33             |
| 2.5V LVC MOS            | LVC MOS25             |
| 1.8V LVC MOS            | LVC MOS18             |
| 1.5V LVC MOS            | LVC MOS15             |
| 1.2V LVC MOS            | LVC MOS12             |
| 3.3V PCI                | PCI33                 |

1. 这些差分标准通过使用有外部电阻组合的互补 LVC MOS 驱动器实现。

## OPENDRAIN

LVC MOS 和 LVTTL I/O 标准可通过 OPENDRAIN 属性设为开漏配置。

属性值: ON, OFF

默认值: OFF

## DRIVE

DRIVE 属性可用于为有可编程驱动能力的输出标准设置可编程驱动强度。



表 8. DRIVE 设置

| 输出标准                  | 驱动 (mA)          | 默认值 (mA) |
|-----------------------|------------------|----------|
| HSTL15_I/ HSTL15D_I   | 4, 8             | 8        |
| HSTL18_I/ HSTL18D_I   | 8, 12            | 12       |
| SSTL25_I/ SSTL25D_I   | 8, 12            | 8        |
| SSTL25_II/ SSTL25D_II | 16, 20           | 16       |
| SSTL18_II/SSTL18D_II  | 8, 12            | 12       |
| LVC MOS12             | 2, 6             | 6        |
| LVC MOS15             | 4, 8             | 8        |
| LVC MOS18             | 4, 8, 12, 16     | 12       |
| LVC MOS25             | 4, 8, 12, 16, 20 | 12       |
| LVC MOS33             | 4, 8, 12, 16, 20 | 12       |
| LV TTL                | 4, 8, 12, 16, 20 | 12       |

### PULLMODE

PULLMODE 属性可用于所有 LV TTL 和 LVC MOS 的输入和输出。该属性可单独针对每个 I/O 进行使能。

属性值: UP, DOWN, NONE, KEEPER

默认值: UP

表 9. PULLMODE 值

| PULL 选项      | PULLMODE 值 |
|--------------|------------|
| Pull-up (默认) | UP         |
| Pull-down    | DOWN       |
| Bus Keeper   | KEEPER     |
| Pull Off     | NONE       |

### PCICLAMP

PCI33 输入位于 LatticeECP2 器件的下侧或者 LatticeECP2M 器件的左侧和下侧, 拥有一个可选的 PCI 钳位二极管, 可通过 PCICLAMP 属性来使能。PCICLAMP 还可用于所有的 LVC MOS33 和 LV TTL 输入。

属性值: ON, OFF

默认值: OFF

表 10. PCICLAMP 值

| 输入类型      | PCICLAMP 值    |
|-----------|---------------|
| PCI33     | ON            |
| LVC MOS33 | OFF (默认值), ON |
| LV TTL    | OFF (默认值), ON |

### SLEWRATE

SLEWRATE 属性可用于所有 LV TTL 和 LVC MOS 输出驱动器。每个 I/O 引脚有其单独的摆率控制。这使得设计师可以对每个引脚进行摆率控制。

属性值: FAST, SLOW

默认值: FAST

## FIXEDEDELAY

每个输入引脚都有 **FIXEDEDELAY** 属性。该属性使能后，当使用全局时钟时，输入寄存器可实现零保持时间。该属性仅可在 HDL 源代码中进行设置。

属性值：TRUE, FALSE

默认值：FALSE

## INBUF

默认情况下，所有未使用的输入缓冲器都禁用。**INBUF** 属性用于在执行边界扫描测试时，使能未使用的输入缓冲器。这是一个全局属性可统一设为 ON 或 OFF。

属性值：ON, OFF

默认值：OFF

## DIN/DOUT

该属性可用于 I/O 寄存器的分配。使用 **DIN** 为设计分配一个输入寄存器，使用 **DOUT** 来分配输出寄存器。默认情况下，软件会分配可用的 I/O 寄存器。用户可以通过使用综合属性或 ispLEVER 软件的 Preference Editor 关闭该功能。这两个属性只能用于寄存器。

## LOC

该属性可用于设计中 I/O 端口的引脚分配。该属性仅在 HDL 源代码中进行引脚分配时使用。还可以直接通过 ispLEVER 软件的 Preference Editor GUI 进行引脚分配。进一步信息请参见附录。

## 设计考虑和使用

本章节讨论了使用 LatticeECP2/M sysIO 缓冲器进行设计时必须考虑的一些规则和考虑。

### Bank 规则

- 如果任何一个 bank 的  $V_{CCIO}$  或  $V_{CCJ}$  设为 3.3V，建议  $V_{CCAUX}$  也连到相同的电源，从而使漏电流最小化。
- 如果任何一个 bank 的  $V_{CCIO}$  或  $V_{CCJ}$  设为 1.2V，建议  $V_{CC}$  也连到相同的电源，从而使漏电流最小化。
- 当实现 DDR 存储器接口时，bank 的  $V_{REF1}$  是用于提供接口引脚的参考电压并且不能对其他任何参考输入进行供电。
- 仅 LatticeECP2 下侧的 bank（Bank 4 和 5）或 LatticeECP2M 左侧和下侧的 bank（Bank 4,5,6 和 7）支持可编程的 PCI 钳位二极管。
- 所有合法的输入缓冲器都独立于 bank 的  $V_{CCIO}$ ，除了 1.8V 和 1.5V 缓冲器，它们要求 bank 的  $V_{CCIO}$  为 1.8V 和 1.5V。

### 差分 I/O 规则

- 所有 bank 都支持 LVDS 输入缓冲器。仅右侧和左侧的 bank（Bank 2、3、6 和 7）可支持真差分输出缓冲器。上侧和下侧的 bank 可支持 LVDS 输入缓冲器，但不支持真 LVDS 输出。用户可在这些 bank 上使用仿真 LVDS 输出缓冲器。
- 所有 bank 都支持使用外部电阻组合和互补 LVCMOS 驱动器的仿真差分缓冲器。
- 左、右侧 bank 中仅 50% 的 I/O 可提供 LVDS 输出缓冲器。LVDS 仅可分配引脚给 TRUE 极。ispLEVER 设计工具会自动将差分对的另一个 I/O 分配给互补极。参见器件数据手册，查看所有 LVDS 对的引脚列表。

## 为参考输入分配 $V_{REF1}/V_{REF2}$ 组

每个 bank 有两个专用的  $V_{REF}$  输入引脚， $V_{REF1}$  和  $V_{REF2}$ 。设计师可将缓冲器按特定的  $V_{REF}$  值， $V_{REF1}$  或  $V_{REF2}$  进行分组。这种分组是通过指定 PGROUP VREF 可选属性和 LOCATE PGROUP 可选属性来实现的。

### 可选属性语法

```
PGROUP <pgrp_name> [(VREF <vref_name>)+] (COMP <comp_name>)+;
LOCATE PGROUP <pgrp_name> BANK <bank_num>;
LOCATE VREF <vref_name> SITE <site_name>;
```

### 显示 VREF 组（VREF Groups）的示例

```
PGROUP "vref_pg1" VREF "ref1" COMP "ah(0)" COMP "ah(1)" COMP "ah(2)" COMP "ah(3)" COMP "ah(4)"
COMP "ah(5)" COMP "ah(6)" COMP "ah(7)";
```

```
PGROUP "vref_pg2" VREF "ref2" COMP "al(0)" COMP "al(1)" COMP "al(2)" COMP "al(3)" COMP "al(4)" COMP
"al(5)" COMP "al(6)" COMP "al(7)";
```

```
LOCATE VREF "ref1" SITE PR29C;
LOCATE VREF "ref2" SITE PR48B;
```

或

```
LOCATE PGROUP "vref_pg1" BANK 2;
LOCATE PGROUP "vref_pg2" BANK 2;
```

示例中的两个  $V_{REF}$  组，“vref\_pg1”分配给 VREF “ref1”和“vref\_pg2”分配给“ref2”。用户必须使用 LOCATE 可选属性将  $V_{REF}$  锁定到  $V_{REF1}$  或  $V_{REF2}$ 。或者，用户可以指定  $V_{REF}$  组分到哪个 bank。然后，软件会将它们分配给该 bank 内的  $V_{REF1}$  或  $V_{REF2}$ 。

如果没有使用 PGROUP VREF，软件会自动将所有需要相同  $V_{REF}$  参考电压的引脚分为一组。在一根以上的总线使用相同参考电压并且用户希望将这些总线连到不同  $V_{REF}$  源时，该可选属性会非常有用。

## 差分 I/O 的实现

LatticeECP2/M 器件支持各种差分标准，详细内容请见下面章节。

### LVDS

真 LVDS（LVDS25）驱动器在器件左、右侧的 50% 的 I/O 上可用。器件的各侧 bank 都提供 LVDS 输入支持。器件的四侧 bank 都使用外部电阻的互补 LVCMOS 驱动器（LVDS25E）支持 LVDS。请参见 [LatticeECP2/M 系列数据手册](#)，了解 LVDS 实现的详细说明。

### BLVDS

所有的单端 sysIO 缓冲器对都使用外部电阻的互补 LVCMOS 驱动器支持 Bus-LVDS 标准。请参见 [LatticeECP2/M 系列数据手册](#)，了解 BLVDS 实现的详细说明。

### RSDS

所有的单端 sysIO 缓冲器对都使用外部电阻的互补 LVCMOS 驱动器支持 RSDS 标准。请参见 [LatticeECP2/M 系列数据手册](#)，了解 RSDS 实现的详细说明。

## LVPECL

所有的 sysIO 缓冲器都支持 LVPECL 输入。使用外部电阻的互补 LVCMOS 驱动器支持 LVPECL 输出。请参见 [LatticeECP2/M 系列数据手册](#)，了解 LVPECL 实现的详细说明。

## 差分 SSTL 和 HSTL

所有的单端 sysIO 缓冲器都支持差分 SSTL 和 HSTL。请参见 [LatticeECP2/M 系列数据手册](#)，了解差分 HSTL 和 SSTL 实现的详细说明。

## 技术支持

热线电话: +86-21-52989090

电子邮件: [techsupport-asia@latticesemi.com](mailto:techsupport-asia@latticesemi.com)

网站: [www.latticesemi.com.cn](http://www.latticesemi.com.cn)

## 修订历史

| 日期         | 版本   | 更改摘要  |
|------------|------|---|
| 2006 年 2 月 | 01.0 | 最初的版本。                                      |
| 2006 年 9 月 | 01.1 | 更新加入了 LatticeECP2M 支持。                      |
| 2007 年 4 月 | 01.2 | 更新了支持的输出标准表。                                |
| 2007 年 6 月 | 01.3 | 更新了用于 DDR 存储器接口的 $V_{REF1}$ 章节。             |
| 2007 年 6 月 | 01.4 | 更新了 Bank 表支持的 sysIO 标准。<br>更新了 Bank 规则要点列表。 |
| 2007 年 6 月 | 01.5 | 更新了电源表。                                     |
| 2008 年 4 月 | 01.6 | 更新了 LatticeECP2M sysIO Bank 图。              |
| 2009 年 2 月 | 01.7 | 更新了 Bank 表支持的 I/O 标准。                       |

## 附录 A. 用于 Synplicity® 和 Precision® RTL 综合的 HDL 属性

设计师们可以直接在 HDL 源代码中指定 sysIO 属性。这必然会用到相关综合工具的属性定义和语法。下面是适用于 Precision RTL Synthesis 和 Synplicity 的 sysIO 属性、语法和示例。本章节仅列出了适用于这些器件的 sysIO 缓冲器属性。您可以参考 Precision RTL Synthesis 和 Synplicity 用户手册，查阅综合属性的完整列表。您还可以通过 ispLEVER 软件的 Help 查看用户手册。

### VHDL Synplicity/Precision RTL 综合

本章节列出了使用 Precision RTL Synthesis 或 Synplicity 综合工具时，在 VHDL 中所有 sysIO 属性语法和示例。

#### 语法

表 11. 用于 Synplicity 和 Precision RTL 综合的 VHDL 属性语法

| 属性         | 语法  |
|------------|---|
| IO_TYPE    | attribute IO_TYPE: string;<br>attribute IO_TYPE of Pinname: signal is "IO_TYPE Value";          |
| OPENDRAIN  | attribute OPENDRAIN: string;<br>attribute OPENDRAIN of Pinname: signal is "OpenDrain Value";    |
| DRIVE      | attribute DRIVE: string;<br>attribute DRIVE of Pinname: signal is "Drive Value";                |
| PULLMODE   | attribute PULLMODE: string;<br>attribute PULLMODE of Pinname: signal is "Pullmode Value";       |
| PCICLAMP   | attribute PCICLAMP: string;<br>attribute PCICLAMP of Pinname: signal is "PCIClamp Value";       |
| SLEWRATE   | attribute PULLMODE: string;<br>attribute PULLMODE of Pinname: signal is "Slewrates Value";      |
| FIXEDDELAY | attribute FIXEDDELAY: string;<br>attribute FIXEDDELAY of Pinname: signal is "Fixeddelay Value"; |
| DIN        | attribute DIN: string;<br>attribute DIN of Pinname: signal is "";                               |
| DOUT       | attribute DOUT: string;<br>attribute DOUT of Pinname: signal is "";                             |
| LOC        | attribute LOC: string;<br>attribute LOC of Pinname: signal is "pin_locations";                  |

#### 示例

##### IO\_TYPE

```
--***Attribute Declaration***
ATTRIBUTE IO_TYPE: string;
--***IO_TYPE assignment for I/O Pin***
ATTRIBUTE IO_TYPE OF portA: SIGNAL IS "PCI33";
ATTRIBUTE IO_TYPE OF portB: SIGNAL IS "LVCMOS33";
ATTRIBUTE IO_TYPE OF portC: SIGNAL IS "LVDS25";
```

##### OPENDRAIN

```
--***Attribute Declaration***
ATTRIBUTE OPENDRAIN: string;
--***DRIVE assignment for I/O Pin***
ATTRIBUTE OPENDRAIN OF portB: SIGNAL IS "ON";
```

##### DRIVE

---

```
--***Attribute Declaration***  
ATTRIBUTE DRIVE: string;  
--***DRIVE assignment for I/O Pin***  
ATTRIBUTE DRIVE OF portB: SIGNAL IS "20";
```

#### PULLMODE

```
--***Attribute Declaration***  
ATTRIBUTE PULLMODE : string;  
--***PULLMODE assignment for I/O Pin***  
ATTRIBUTE PULLMODE OF portA: SIGNAL IS "DOWN";  
ATTRIBUTE PULLMODE OF portB: SIGNAL IS "UP";
```

#### PCICLAMP

```
--***Attribute Declaration***  
ATTRIBUTE PCICLAMP: string;  
--***PULLMODE assignment for I/O Pin***  
ATTRIBUTE PCICLAMP OF portA: SIGNAL IS "ON";
```

#### SLEWRATE

```
--***Attribute Declaration***  
ATTRIBUTE SLEWRATE : string;  
--*** SLEWRATE assignment for I/O Pin***  
ATTRIBUTE SLEWRATE OF portB: SIGNAL IS "FAST";
```

#### FIXEDDELAY

```
--***Attribute Declaration***  
ATTRIBUTE FIXEDDELAY: string;  
--*** SLEWRATE assignment for I/O Pin***  
ATTRIBUTE FIXEDDELAY OF portB: SIGNAL IS "TRUE";
```

#### DIN/DOUT

```
--***Attribute Declaration***  
ATTRIBUTE din : string;  
ATTRIBUTE dout : string;  
--*** din/dout assignment for I/O Pin***  
ATTRIBUTE din OF input_vector: SIGNAL IS "";  
ATTRIBUTE dout OF output_vector: SIGNAL IS "";
```

#### LOC

```
--***Attribute Declaration***  
ATTRIBUTE LOC : string;  
--*** LOC assignment for I/O Pin***  
ATTRIBUTE LOC OF input_vector: SIGNAL IS "E3,B3,C3 "";
```

## Verilog Synplicity

本章节列出了使用 Synplicity 综合工具时，在 Verilog 中所有 sysIO 属性语法和示例。

### 语法

表 12. Verilog Synplicity 的属性语法

| 属性        | 语法   |
|-----------|--|
| IO_TYPE   | <i>PinType PinName</i> /* synthesis IO_TYPE="IO_Type Value"*/;       |
| OPENDRAIN | <i>PinType PinName</i> /* synthesis OPENDRAIN ="OpenDrain Value"*/;  |
| DRIVE     | <i>PinType PinName</i> /* synthesis DRIVE="Drive Value"*/;           |
| PULLMODE  | <i>PinType PinName</i> /* synthesis PULLMODE="Pullmode Value"*/;     |
| PCICLAMP  | <i>PinType PinName</i> /* synthesis PCICLAMP =" PCIClamp Value"*/;   |
| SLEWRATE  | <i>PinType PinName</i> /* synthesis SLEWRATE="Slewrates Value"*/;    |
| FIXEDELAY | <i>PinType PinName</i> /* synthesis FIXEDDELAY="Fixeddelay Value"*/; |
| DIN       | <i>PinType PinName</i> /* synthesis DIN=" "*/;                       |
| DOUT      | <i>PinType PinName</i> /* synthesis DOUT=" "*/;                      |
| LOC       | <i>PinType PinName</i> /* synthesis LOC="pin_locations "*/;          |

### 示例

#### //IO\_TYPE, PULLMODE, SLEWRATE and DRIVE assignment

```
output portB /*synthesis IO_TYPE="LVCMOS33" PULLMODE ="UP" SLEWRATE ="FAST"
DRIVE ="20"*/;
output portC /*synthesis IO_TYPE="LVDS25" */;
```

#### //OPENDRAIN

```
output portA /*synthesis OPENDRAIN ="ON"*/;
```

#### //PCICLAMP

```
output portA /*synthesis IO_TYPE="PCI33" PULLMODE ="PCICLAMP"*/;
```

#### // Fixeddelay

```
input load /* synthesis FIXEDDELAY="TRUE" */;
```

#### // Place the flip-flops near the load input

```
input load /* synthesis din=" " */;
```

#### // Place the flip-flops near the outload output

```
output outload /* synthesis dout=" " */;
```

#### //I/O pin location

```
input [3:0] DATA0 /* synthesis loc="E3,B1,F3"*/;
```

#### //Register pin location

```
reg data_in_ch1_buf_reg3 /* synthesis loc="R40C47" */;
```

#### //Vectored internal bus

```
reg [3:0] data_in_ch1_reg /*synthesis loc ="R40C47,R40C46,R40C45,R40C44" */;
```

## Verilog Precision

本章节列出了使用 Precision RTL Synthesis 综合工具时，在 Verilog 中所有 sysIO 属性语法和示例。

### 语法

表 13. Verilog Precision 的属性语法

| 属性         | 语法   |
|------------|--|
| IO_TYPE    | //pragma attribute <i>PinName</i> IO_TYPE IO_TYPE Value      |
| OPENDRAIN  | // pragma attribute <i>PinName</i> OPENDRAIN OpenDrain Value |
| DRIVE      | // pragma attribute <i>PinName</i> DRIVE Drive Value         |
| PULLMODE   | // pragma attribute <i>PinName</i> IO_TYPE Pullmode Value    |
| PCICLAMP   | // pragma attribute <i>PinName</i> PCICLAMP PCIClamp Value   |
| SLEWRATE   | // pragma attribute <i>PinName</i> IO_TYPE Slewrate Value    |
| FIXEDDELAY | // pragma attribute <i>PinName</i> IO_TYPE Fixeddelay Value  |
| LOC        | // pragma attribute <i>PinName</i> LOC pin_location          |

### 示例

```

*****IO_TYPE ***
//pragma attribute portA IO_TYPE PCI33
//pragma attribute portB IO_TYPE LVCMOS33
//pragma attribute portC IO_TYPE SSTL25_II

**** Opendrain ***
//pragma attribute portB OPENDRAIN ON
//pragma attribute portD OPENDRAIN OFF

**** Drive ***
//pragma attribute portB DRIVE 20
//pragma attribute portD DRIVE 8

**** Pullmode***
//pragma attribute portB PULLMODE UP

**** PCIClamp***
//pragma attribute portB PCICLAMP ON

**** Slewrate ***
//pragma attribute portB SLEWRATE FAST
//pragma attribute portD SLEWRATE SLOW

// ***Fixeddelay***
// pragma attribute load FIXEDDELAY TRUE

****LOC***
//pragma attribute portB loc E3
    
```



## 附录 B. 使用 Preference Editor 用户界面的 sysIO 属性

设计师可以通过 ispLEVER 设计工具中的 Pre-Map Preference Editor GUI 来指定 sysIO 缓冲器属性。Pin Attribute 表列出了设计中的所有端口和所有可用的 sysIO 可选属性。点击每个单元格，查看该端口的所有有效的可选 I/O 可选属性。每一列的可选项由前一列所决定。因此，当您选择一个特定的 IO\_TYPE, DRIVE、PULLMODE 和 SLEW-RATE 列的属性将只列出适用于该 IO\_TYPE 的有效属性组合。引脚位置可通过 Pin Attribute 表的 Pin Location 列锁定。右击单元格列出所有可用的引脚位置。Preference Editor 还将执行 DRC 检查，以确定是否有不正确的引脚分配。

设计师可以通过 Preference Editor 的 Cell Attributes 表输入 DIN/ DOUT 的可选值。所有通过 Preference Editor 分配的可选项都会写入 preference 文件 (.prf)。

图 2 和图 3 展示了 Preference Editor 中 Pin Attribute 表和 Cell Attribute 表。有关如何使用 Preference Editor 的进一步信息，请参考软件 Help 菜单选项下的 ispLEVER Help 文档。

图 2. Pin Attributes 选项卡

| Type | Signal/Gr... | Groupe... | Pin Location | IO Type | Drive       | Slewrate | Pullmode | Output Load |
|------|--------------|-----------|--------------|---------|-------------|----------|----------|-------------|
| 2    | Output Port  | portD(3)  | N/A          |         | N/A         | N/A      | N/A      |             |
| 3    | Output Port  | portD(2)  | N/A          |         | N/A         | N/A      | N/A      |             |
| 4    | Output Port  | portD(1)  | N/A          |         | N/A         | N/A      | N/A      |             |
| 5    | Output Port  | portD(0)  | N/A          |         | N/A         | N/A      | N/A      |             |
| 6    | Output Port  | portC(4)  | N/A          | A17     | LVCMS533    |          | NONE     |             |
| 7    | Output Port  | portC(3)  | N/A          | A18     | BLVDS25     |          | NONE     | N/A         |
| 8    | Output Port  | portC(2)  | N/A          | A19     | LVCMS525_OD |          | NONE     |             |
| 9    | Output Port  | portC(1)  | N/A          | A20     | LVCMS515    |          | NONE     |             |
| 10   | Output Port  | portC(0)  | N/A          | A15     | LVPECL33    |          | NONE     | N/A         |
| 11   | Output Port  | portB(4)  | N/A          |         |             | N/A      | N/A      |             |
| 12   | Output Port  | portB(3)  | N/A          |         |             | N/A      | N/A      |             |
| 13   | Output Port  | portB(2)  | N/A          |         |             | N/A      | N/A      |             |
| 14   | Output Port  | portB(1)  | N/A          |         |             | N/A      | N/A      |             |

图 3. Cell Attributes 选项卡

| Type | Cell Name | Din / Dout |      |
|------|-----------|------------|------|
| 1    | FFs       | ix266      | Din  |
| 2    | FFs       | ix205      | Din  |
| 3    | FFs       | ix212      | Din  |
| 4    | FFs       | ix215      | Din  |
| 5    | FFs       | ix218      | Din  |
| 6    | FFs       | ix221      | Din  |
| 7    | FFs       | ix224      | Dout |
| 8    | FFs       | ix227      | Dout |
| 9    | FFs       | ix230      | Dout |
| 10   | FFs       | ix233      | Din  |
| 11   | FFs       | ix236      | Dout |
| 12   | FFs       | ix239      | Din  |
| 13   | FFs       | ix242      | Din  |

## 附录 C. 使用 Preference File（ASCII 文件）的 sysIO 属性

设计师可以直接在 preference 文件（.prf）中输入 sysIO 属性作为 sysIO 缓冲器的可选属性。PRF 文件是一个 ASCII 文件，包含两个独立部分：一个是由映射器或转换器生成的可选属性的原理图部分，以及一个用户输入的可选属性部分。用户的可选属性可直接写入这个文件。综合属性出现在文件中原理图的起始和结尾之间。sysIO 缓冲器可选属性可在原理图定义块之后按照 preference 文件语法进行输入。下面是一系列 sysIO 缓冲器可选参数的语法和示例。

### IOBUF

该可选属性是用来设置属性 IO\_TYPE、PULLMODE、SLEWRATE 和 DRIVE。

#### 语法

```
IOBUF [ALLPORTS | PORT <port_name> | GROUP <group_name>] (keyword=<value>)+;
```

其中：

<port\_name> = 这不是实际最上层的端口名称，而是该端口的信号名称。物理设计（.ncd）文件中的 PIO 使用这种规则命名。多个端口名称列表或带有通配符的端口名称应使用 GROUP 来定义。

Keyword = IO\_TYPE, OPENDRAIN, DRIVE, PULLMODE, PCICLAMP, SLEWRATE.

#### 示例

```
IOBUF PORT "port1" IO_TYPE=LVTTL33 OPENDRAIN=ON DRIVE=8 PULLMODE=UP
PCICLAMP =OFF SLEWRATE=FAST;
DEFINE GROUP "bank1" "in*" "out_[0-31]";
IOBUF GROUP "bank1" IO_TYPE=SSTL18_II;
```

### LOCATE

当这个可选属性用于特定元件时，它将元件置于特定位置并进行锁定。如果该属性用于特定宏实例，他将宏参考元件置于特定位置，并将宏实例的所有预置元件（即：宏实例库文件中的所有元件）置于参考元件的相应位置，并且对所有放置的元件进行锁定。该属性也可以用于特定的 PGROUP。

#### 语法

```
LOCATE [COMP <comp_name> | MACRO <macro_name>] SITE <site_name>;
LOCATE PGROUP <pgroup_name> [SITE <site_name>; | REGION <region_name>;]
LOCATE PGROUP <pgroup_name> RANGE <site_1> [<site_2> | <count>] [<direction>] | RANGE <chip_side>
[<direction>];
LOCATE BUS <bus_name> ROW|COL <number>;
<bus_name> := string
<number> := integer
```

注：如果 comp\_name、macro\_name 或 site\_name 用除 alpha 字符（如：“11C7”）以外的其他字符开始，你必须在名称外加双引号。<comp\_name> 可使用通配符。

#### 示例

这个指令将端口 Clk0 放在名为 A4 的位置中：

```
LOCATE COMP "Clk0" SITE "A4";
```

这个命令将元件 PFU1 放在名为 R1C7 的位置中：

```
LOCATE COMP "PFU1" SITE "R1C7";
```

这个指令将 bus1 放在 ROW 3 并将 bus2 放在 COL4

```
LOCATE BUS "bus1" ROW 3;  
LOCATE BUS "bus2" COL 4;
```

## USE DIN CELL

这个可选属性指定了给定的寄存器用作一个输入触发器。

### 语法

```
USE DIN CELL <cell_name>;
```

其中:

```
<cell_name> := string
```

### 示例

```
USE DIN CELL "din0";
```

## USE DOUT CELL

该属性指定了给定的寄存器用作一个输出触发器。

### 语法

```
USE DOUT CELL <cell_name>;
```

其中:

```
<cell_name> := string
```

### 示例

```
USE DOUT CELL "dout1";
```

## PGROUP VREF

这个可选属性适用于将所有需要连到一个 bank 内的同一个  $V_{REF}$  引脚的元件进行分组。

### 语法

```
PGROUP <pgrp_name> [(VREF <vref_name>)+] (COMP <comp_name>+);  
LOCATE PGROUP <pgrp_name> BANK <bank_num>;  
LOCATE VREF <vref_name> SITE <site_name>;
```

### 示例

```
PGROUP "vref_pg1" VREF "ref1" COMP "ah(0)" COMP "ah(1)" COMP "ah(2)" COMP "ah(3)" COMP "ah(4)"  
COMP "ah(5)" COMP "ah(6)" COMP "ah(7)";  
PGROUP "vref_pg2" VREF "ref2" COMP "al(0)" COMP "al(1)" COMP "al(2)" COMP "al(3)" COMP "al(4)" COMP  
"al(5)" COMP "al(6)" COMP "al(7)";  
LOCATE VREF "ref1" SITE PR29C;  
LOCATE VREF "ref2" SITE PR48B;
```

或:

```
LOCATE PGROUP "vref_pg1" BANK 2;  
LOCATE PGROUP "vref_pg2" BANK 2;
```