

## MIPI D-PHY 参数配置指南

### 审 批

	签名	日期
拟制人/部门	耿阿囡	2017-1-3
审核		

# Contents

<b>1</b>	<b>文档介绍.....</b>	<b>3</b>
1.1	文档目的.....	3
1.2	参考文档.....	3
1.3	术语与缩写解释 .....	3
<b>2</b>	<b>配置说明.....</b>	<b>3</b>
2.1	技术背景.....	3
2.2	D-PHY 参数定义 .....	3
2.3	D-PHY 参数计算 .....	6
2.3.1	DSI 模块.....	6
2.3.2	CSI 模块.....	6
<b>3</b>	<b>参数修改说明.....</b>	<b>6</b>
<b>4</b>	<b>驱动调试指南.....</b>	<b>7</b>
4.1	CSI 模块 .....	7
4.2	数据 Lane Mapping .....	8

# 1 文档介绍

## 1.1 文档目的

本文档为 SD 和 FW 人员介绍 MIPI D-PHY 参数配置，适用于目前所有 IC 的 MIPI CSI 和 DSI 模块。

## 1.2 参考文档

Specification for D-PHY, V1.2

## 1.3 术语与缩写解释

缩写、术语	解 释
DSI	Display Serial Interface
CSI	Camera Serial Interface

# 2 配置说明

## 2.1 技术背景

MIPI 作为一种高速串行接口，其广泛用于便携式设备中。目前我们常用的有两种：DSI 和 CSI。DSI 是显示设备接口，用于高清显示屏的接口，驱动方称为 DSI Tx 设备，屏称为 DSI Rx 设备；CSI 是 camera 传输接口，用于高清摄像头的连接（我们目前实现的是 CSI-2 标准），Camera 方称为 CSI Tx 方，接收方称为 CSI Rx 方。所以，在我们公司目前 IC 中，DSI 和 CSI 模块分别扮演的是 DSI Tx 方和 CSI Rx 方。DSI 和 CSI 只是协议部分的区别，两种标准在模拟部分传输时均采用 D-PHY 标准。所以清楚的了解 D-PHY 标准，对于调试 DSI 和 CSI 模块是至关重要的。

MIPI 标准定义中，定义了两种传输方式，高速 HS 和低功耗 LP 模式。简言之，就是在传输高速信号的时候，切换到 HS 模式（电平 Max0.36V），在传输完毕后，切换到 LP 模式（电平 Normal1.2V）。在传输过程中，就是有一系列的 HS-LP、LP-HS 的转换过程。在每次转换中，CSI/DSI 的 Tx 端需要按照标准的 timing 信息来发送信息，便于 CSI/DSI Rx 端还原信息。当检测到有不符合标准的时序时，就会用不同的 error 信息反馈到控制器中便于用户调试。

## 2.2 D-PHY 参数定义

MIPI 信号通常包括一组 clock lane 和一到四组 data lane。图 1 和表 1、图 2 和表 2 分别表现 MIPI D-PHY

Clock lane 和 Data lane 的时序转换关系。

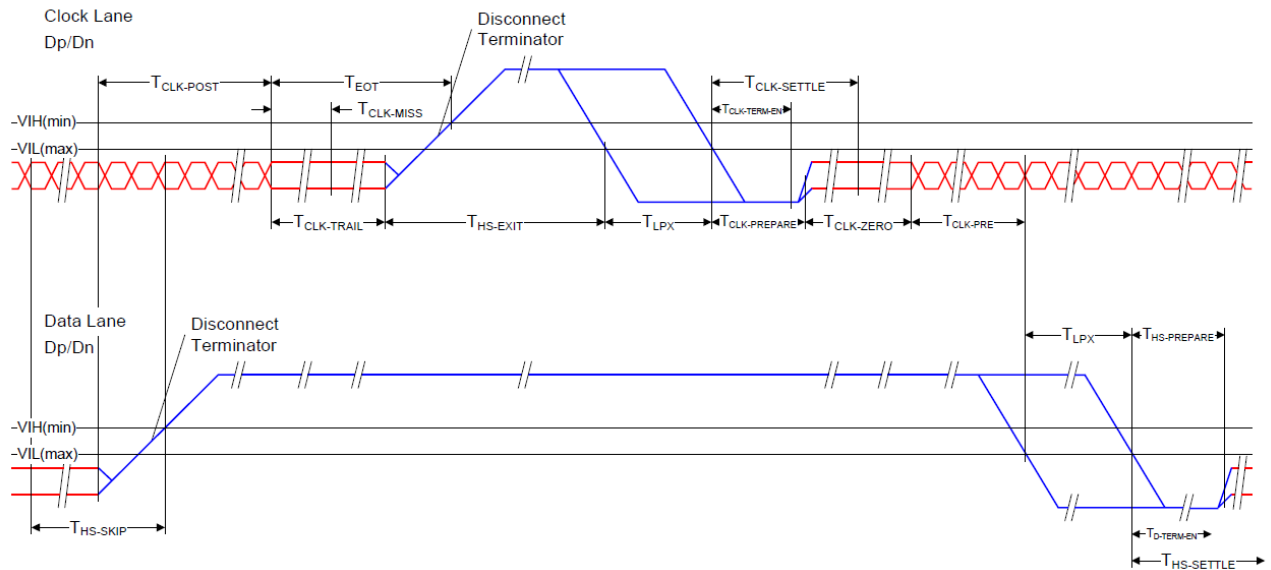


图 1 Clock Lane 转换时序图

表 1 Clock Lane 关键参数

Parameter	Description	Min	Typ	Max	Unit	Notes
T <sub>CLK-POST</sub>	Time that the transmitter continues to send HS clock after the last associated Data Lane has transitioned to LP Mode. Interval is defined as the period from the end of T <sub>HS-TRAIL</sub> to the beginning of T <sub>CLK-TRAIL</sub> .	60 ns + 52*UI			ns	Tx
T <sub>CLK-PRE</sub>	Time that the HS clock shall be driven by the transmitter prior to any associated Data Lane beginning the transition from LP to HS mode.	8			UI	Tx
T <sub>CLK-PREPARE</sub>	Time that the transmitter drives the Clock Lane LP-00 Line state immediately before the HS-0 Line state starting the HS transmission.	38		95	ns	Tx
T <sub>CLK-SETTLE</sub>	Time interval during which the HS receiver should ignore any Clock Lane HS transitions, starting from the beginning of T <sub>CLK-PREPARE</sub> .	95		300	ns	Rx
T <sub>CLK-TERM-EN</sub>	Time for the Clock Lane receiver to enable the HS line termination, starting from the time point when Dn crosses V <sub>ILMAX</sub> .	Time for Dn to reach V <sub>TERM-EN</sub>		38	ns	Rx
T <sub>CLK-TRAIL</sub>	Time that the transmitter drives the HS-0 state after the last payload clock bit of a HS transmission burst.	60			ns	Tx
T <sub>CLK-PREPARE</sub> + T <sub>CLK-ZERO</sub>	T <sub>CLK-PREPARE</sub> + time that the transmitter drives the HS-0 state prior to starting the Clock.	300			ns	Tx
T <sub>LPX</sub>	Transmitted length of any Low-Power state period	50			ns	Tx

图 1 和表 1 中有几个重要参数需要注意下：

- 1) Rx 端的 Tclk\_term\_en 必须小于 Tx 端的 Tclk\_prepare, 即  $T_{clk\_term\_en} < T_{clk\_prepare}$ ;
- 2) Rx 端的 Tclk\_settle 必须小于 Tx 端的  $T_{clk\_prepare} + T_{clk\_zero}$ , 即  $T_{clk\_settle} < (T_{clk\_prepare} + T_{clk\_zero})$ 。

这两个参数配置不合适将会导致出现接收方接收错误, 所以对于我们 DSI Tx 端来说, 必须按照上表配置 Tclk\_prepare 和 Tclk\_zero 参数。

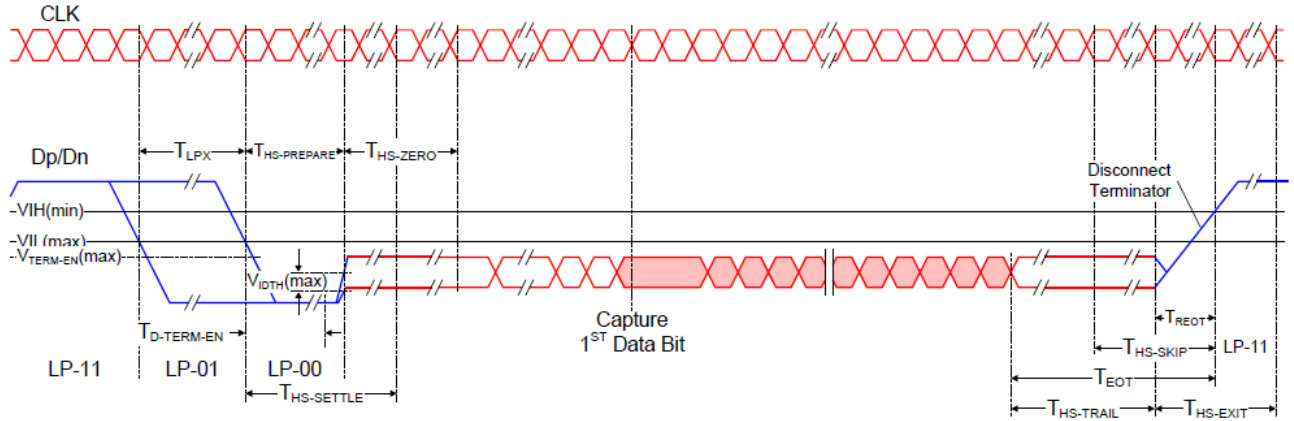


图 2 Data Lane 转换时序图

表 2 Data Lane 关键参数

Parameter	Description	Min	Typ	Max	Unit	Notes
$T_{D-TERM-EN}$	Time for the Data Lane receiver to enable the HS line termination, starting from the time point when Dn crosses $V_{ILMAX}$ .	Time for Dn to reach $V_{TERM-EN}$		$35\text{ ns} + 4 \cdot UI$		Rx
$T_{HS-EXIT}$	Time that the transmitter drives LP-11 following a HS burst.	100			ns	Tx
$T_{HS-PREPARE}$	Time that the transmitter drives the Data Lane LP-00 Line state immediately before the HS-0 Line state starting the HS transmission	$40\text{ ns} + 4 \cdot UI$		$85\text{ ns} + 6 \cdot UI$	ns	Tx
$T_{HS-PREPARE} + T_{HS-ZERO}$	$T_{HS-PREPARE}$ + time that the transmitter drives the HS-0 state prior to transmitting the Sync sequence.	$145\text{ ns} + 10 \cdot UI$			ns	Tx
$T_{HS-SETTLE}$	Time interval during which the HS receiver shall ignore any Data Lane HS transitions, starting from the beginning of $T_{HS-PREPARE}$ . The HS receiver shall ignore any Data Lane transitions before the minimum value, and the HS receiver shall respond to any Data Lane transitions after the maximum value.	$85\text{ ns} + 6 \cdot UI$		$145\text{ ns} + 10 \cdot UI$	ns	Rx
$T_{HS-SKIP}$	Time interval during which the HS-RX should ignore any transitions on the Data Lane, following a HS burst. The end point of the interval is defined as the beginning of the LP-11 state following the HS burst.	40		$55\text{ ns} + 4 \cdot UI$	ns	Rx
$T_{HS-TRAIL}$	Time that the transmitter drives the flipped differential state after last payload data bit of a HS transmission burst	$\max(n \cdot 8 \cdot UI, 60\text{ ns} + n \cdot 4 \cdot UI)$			ns	Tx $n=1$
$T_{INIT}$	See Section 6.11.	100			$\mu\text{s}$	Tx
$T_{LPX}$	Transmitted length of any Low-Power state period	50			ns	Tx

图 2 和表 2 中有几个重要参数需要注意下：

- 1) Rx 端的  $T_{d\_term\_en}$  必须小于 Tx 端的  $T_{hs\_prepare}$ ，即  $T_{d\_term\_en} < T_{hs\_prepare}$ ；
- 2) Rx 端的  $T_{hs\_settle}$  必须小于 Tx 端的  $T_{hs\_prepare} + T_{hs\_zero}$ ，即  $T_{hs\_settle} < (T_{hs\_prepare} + T_{hs\_zero})$ 。

这两个参数配置不合适将会导致出现接收方接收错误，所以对于我们 CSI Rx 端来说，必须按照上表配置  $T_{d\_term\_en}$  和  $T_{hs\_settle}$  参数。

备注：

- 1) 表 1 和表 2 中分别用 Tx 和 Rx 标识出参数的适用方，根据设备类型对应设置；
- 2) UI 即传输速率的倒数，比如 MIPI 支持最低的传输速率是 80Mbps，所以对应  $UI = 1/80M = 12.5\text{ns}$ 。

## 2.3 D-PHY 参数计算

### 2.3.1 DSI 模块

IC 设计中, DSI Tx 端所需要的 timing 参数需要通过寄存器进行配置, 包括 DSI\_PHY\_T0 (Clock Lane)、DSI\_PHY\_T1 (Data Lane) 和 DSI\_PHY\_T2 (通用), 这些参数均基于 PHY\_CLK。

下面用 Tclk\_prepare 的计算来举例说明:

假设 PHY\_CLK=40MHz (25ns), 查表 1 得 Tclk\_prepare 的范围是 38ns-95ns, DSI spec 中 Tclk\_prepare 定义是 1/2/4/8/12/16 倍的 Tphy\_clk, 所以选择 2 倍 Tphy\_clk 可以满足 MIPI D-PHY 要求, 而选择 1 则会偏小, 而选择 4 及其以上偏大。

DSI Rx 端的参数定义请参考显示端具体定义, 在此不再涉及, 但要了解对方设置的 Tclk\_term\_en、Tclk\_settle、Td\_term\_en 和 Ths\_settle, 确保我们 DSI Tx 端配置的参数要满足:  $Tclk\_prepare > Tclk\_term\_en$  且  $(Tclk\_prepare + Tclk\_zero) > Tclk\_settle$ 、 $Ths\_prepare > Ths\_term\_en$  且  $(Ths\_prepare + Ths\_zero) > Ths\_settle$ 。

### 2.3.2 CSI 模块

IC 设计中, CSI Rx 端所需要的 timing 参数需要通过寄存器进行配置, 包括 CSIx\_PHY\_T0、CSIx\_PHY\_T1 (Clock Lane) 和 DSI\_PHY\_T2 (Data Lane), 这些参数均基于 CSI\_CLK。

下面用 Td\_term\_en 的计算来举例说明:

假设 CSI\_CLK=216MHz (对应 Tcsi\_clk=4.6ns), 查表 2 得 Td\_term\_en max 是 (35+4UI) ns, 假定选择 30ns, 则按照 spec 中定义  $Td\_term\_en = (Nd\_term\_en + 5) * Tcsi\_clk = 30ns$ , 得到  $Nd\_term\_en = 1$ , 即实际设置的 Td\_term\_en=27.6ns, 符合 MIPI D-PHY 要求。

在此需要说明的是, IC spec 在此部分均有误, spec 中此部分的目前定义可能是 +1, 而实际设计是 +5, 所以计算参数时需要更正过来, 否则就会导致我们实际的 term\_en 和 settle 时间偏大, 如果大于 Tx 端的 prepare 以及 prepare+zero, 就会导致波形出现异常, 更有甚者导致接收出错, 带来兼容性问题。

CSI Tx 端的参数定义请参考 camera 具体定义, 在此不再涉及, 但要了解对方设置的 Tclk\_prepare、Tclk\_zero、Ths\_prepare 和 Ths\_zero, 确保我们 CSI Rx 端配置的参数要满足:  $Tclk\_term\_en < Tclk\_prepare$  且  $Tclk\_settle < (Tclk\_prepare + Tclk\_zero)$ 、 $Ths\_term\_en < Ths\_prepare$  且  $Ths\_settle < (Ths\_prepare + Ths\_zero)$ 。

## 3 参数修改说明

以往驱动代码里面, 是通过外围工具来计算 DSI 和 CSI 这些参数, 然后将参数写入驱动代码中。个人感觉这种方式在后期的维护中不是很好用, 而且也不便于调试。

比如 CSI 模块, 大多数在调试过程中, 可能会直接将 csi\_clk 进行调整, 但往往忘记了同步去更新 timing 的数值。而且, 此软件中计算出来的 timing 信息是根据理论计算出来的 csi\_clk 得到的, 但实际方案中的 csi\_clk 与其不能完全一致, 所以 timing 信息的不准确也会对实际调试带来影响。所以我的建议是像 CSI 这种代码量很小的场合, 直接在驱动里面计算 timing 信息会更为准确且便于维护。

推荐的 CSI Rx 端参数设置以及代码计算方式如下, 推荐参数中 settle 时间是 120ns, 要求 MIPI 传输速率在 200Mbps 之上, 如果在 200Mbps 之下, 则可以适当再将其提高些 (但实际应用场合传输速率都是在 200Mbps 之上)。

```
#define CSI_T_CLK_TERM_EN          30      //单位是 ns
#define CSI_T_CLK_SETTLE          120      //单位是 ns
#define CSI_T_DATA_TERM_EN        30      //单位是 ns
#define CSI_T_DATA_SETTLE         120      //单位是 ns

clk_term_time = CSI_T_CLK_TERM_EN*csi_clk/1000; //csi_clk 单位是 MHz
if (clk_term_time<5)
    clk_term_time = 0;
else
    clk_term_time = clk_term_time - 5;

clk_settle_time = CSI_T_CLK_SETTLE*csi_clk/1000;
if (clk_settle_time<5)
    clk_settle_time = 0;
else
    clk_settle_time = clk_settle_time - 5;

data_term_time = CSI_T_DATA_TERM_EN*csi_clk/1000;
if (data_term_time<5)
    data_term_time = 0;
else
    data_term_time = data_term_time - 5;

data_settle_time = CSI_T_DATA_SETTLE*csi_clk/1000;
if (data_settle_time<5)
    data_settle_time = 0;
else
    data_settle_time = data_settle_time - 5;
IO_WRITEU32(CSIx_PHY_T1, clk_term_time + (clk_settle_time<<4));
IO_WRITEU32(CSIx_PHY_T2, data_term_time + (data_settle_time<<4));
```

## 4 驱动调试指南

### 4.1 CSI 模块

针对 MIPI CSI 接收模块来说，由于要搭配各种 CSI 发送端，所以在驱动调试中需要结合 CSI 发送端具体的信号波形来进行对应配置，大致需要注意如下几个方面：

(1) LP11 电平是否在 0.9V 之上。

目前芯片设计中，MIPI CSI 接收方通过 LP11-LP01-LP00 的电平转换过程来开启高速接收模式，通过 LP00-LP11 的电平变化来结束高速接收模式。如果 LP11 电平偏低，会导致接收方一直按照高速模式接收，

导致接收出错。实际调试中发现，有些 CSI 发送方可通过配置设置为低功耗 LP 模式，所以在出现问题的时候需要将发送方此配置修改为正常 LP 模式。另外需要注意的是，LP11-LP00 以及 LP00-LP11 的电平转换过程在 Data Lane 上面是一定存在的，但在 clock lane 上面可能存在也可能不存在。

(2) Clock Lane 工作在非连续模式 or 连续模式。

Clock Lane 上面有 LP11-LP00 以及 LP00-LP11 的电平转换过程称为非连续模式，否则为连续模式。调试过程中需要用示波器观测 CSI clock lane 的波形，对应设置我们 CSI 接收方的 hclk\_om\_ent\_en 配置 (CSI\_CTRL bit8)，否则会容易导致出错。

(3) 正确配置 CSI\_PHY\_T1 和 CSI\_PHY\_T2 寄存器。

必须正确配置这两个寄存器中的 Settle 和 Term\_en 信息，该信息基于 csi\_clock。具体配置要求请参见第 2.3 章节，具体配置说明请参见第 3 章节描述。

(4) 正确解读 Clock Lane 工作速率以及配置 CSI 模块工作时钟 CSI\_Clock。

Clock Lane 的工作速率是由 MIPI CSI 发送方发送的数据量来决定的，比如 1920x1080、60Hz、YUV422，则总的的数据量大致在 1920x1080x60x16（此处未考虑消隐期，一般需要预留 10%左右的消隐期时间），即至少在 2Gbps，而我们芯片中 CSI 每条 lane 最多工作在 1Gbps，所以需要 3 或者 4 条数据 lane 才能完成传输。假设是 4lane，则每条数据 lane 的传输速率至少在 2Gbps/4=500Mbps，Clock lane 上面测试得到的频率至少为 500Mbps/2=250MHz。所以，通过用示波器测试 clock lane 上的时钟频率可以大致核对下 CSI 发送端是否设置正常。在此，由于一般 CSI 发送端的驱动都是对方提供的，所以 CSI 发送端的配置一般不会有误。而我们则需要通过 Clock Lane 的工作速率来确定 CSI、SI/ISP 模块的工作时钟。比如，此例中，实测 Clock Lane 上面的时钟是 300MHz，则对于 4lane 来说，总的的数据量在 300x2x4=2.4Gbps，由于我们芯片内部 CSI 模块到 SI 或者 ISP 模块的接口是 16bit，所以 CSI、SI/ISP 模块的工作时钟至少需要在 2.4Gbps/16bit=150MHz。但是，对于 RAW10、RAW12 数据来说，其通过 16bit 传输一个 10bit 或者 12bit，所以计算时需要按照具体的有效位来计算，比如对于 RAW10 来说，则 CSI、SI/ISP 模块的工作时钟至少需要在 2.4Gbps/10bit=240MHz。

## 4.2 数据 Lane Mapping

在目前项目中，为了方便 PCB 布线，在 CSI 和 DSI 模块内部均增加了针对 4 条数据 Lane 的通道 Mapping 功能。但 CSI 和 DSI 的 Mapping 功能在具体定义时却有些不同，比较容易导致混淆，在此做进一步的说明。

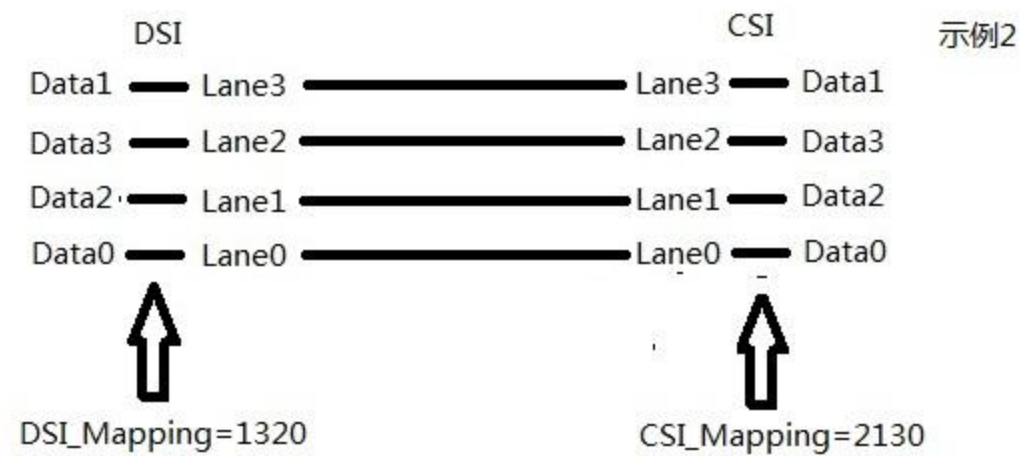
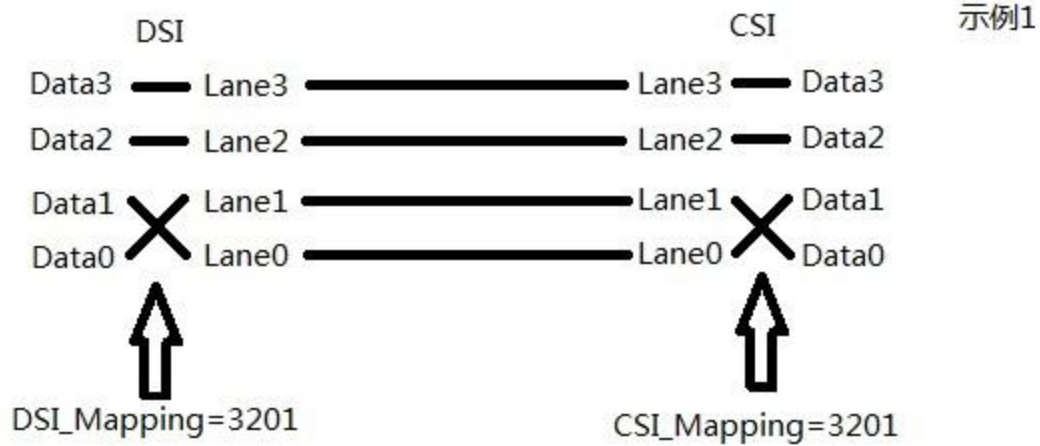
在芯片外围引脚处，CSI 和 DSI 的引脚分别称为 Lane0、Lane1、Lane2、Lane3，每条 Lane 可以通过寄存器 Mapping 功能传输不同的 Data 信息。

DSI\_Mapping = Lane3: Lane2: Lane1: Lane0=DataA: DataB: DataC: DataD，即 LaneX 上面传输的是 DataY。

CSI\_Mapping = Data3: Data2: Data1: Data0= LaneA: LaneB: LaneC: LaneD，即 DataX 通过 LaneY 进行传输。

下面为具体示例图。





DSI\_Mapping  
 =Lane3:Lane2:Lane1:Lane0  
 =DataA:DataB:DataC:DataD  
 DSI\_Mapping是指：  
 LaneX上传输DataY

CSI\_Mapping  
 =Data3:Data2:Data1:Data0  
 =LaneA:LaneB:LaneC:LaneD  
 CSI\_Mapping是指：  
 DataX在LaneY上传输