

## 深入理解 ADAS FPD-LINK III 串行解串器

Jiawei Hu

Sales and Marketing/China Auto Car OEM

### ABSTRACT

FPD-LINK III 作为 TI 汽车级高速 SERDES 产品，广泛应用于车载娱乐系统（IVI）和自动驾驶应用（ADAS）中。在自动驾驶应用中，通常需要传输各种分辨率、接口摄像头数据，因此存在各种分辨率，不同接口 FPD-LINK III 产品搭配使用场景。本应用手册主要帮助用户了解自动驾驶应用中车载摄像头数据传输常见接口，不同接口 FPD-LINK III 串行解串器工作原理以及最后给出不同接口 ADAS FPD-LINK III 产品搭配使用示例代码，帮助用户缩短调试时间，设计出更加稳定的系统。

### Contents

<b>1</b>	<b>车载摄像头数据传输常见接口 .....</b>	<b>2</b>
1.1	DVP 接口 .....	2
1.2	CSI-2 接口 .....	2
<b>2</b>	<b>ADAS FPD-LINK III 串行器介绍 .....</b>	<b>3</b>
2.1	DVP 接口串行器工作原理 .....	4
2.1.1	外部晶振作为参考时钟 .....	5
2.1.2	图像传感器输出 PCLK 作为参考时钟 .....	5
2.1.3	两种参考时钟模式优劣势对比 .....	6
2.2	CSI-2 接口串行器工作原理 .....	7
<b>3</b>	<b>ADAS FPD-LINK III 解串器介绍 .....</b>	<b>8</b>
3.1	DVP 接口解串器工作原理 .....	8
3.2	CSI-2 接口解串器工作原理 .....	9
<b>4</b>	<b>ADAS FPD-LINK III 示例代码 .....</b>	<b>10</b>
4.1	DVP 接口串行器搭配 DVP 接口解串器 .....	10
4.2	DVP 接口串行器搭配 CSI-2 接口解串器 .....	10
4.3	CSI-2 接口串行器搭配 CSI-2 接口解串器 .....	11
4.4	CSI-2 接口串行器搭配 DVP 接口解串器 .....	12
<b>5</b>	<b>参考文献 .....</b>	<b>12</b>

### Figures

Figure 1.	CSI-2 数据差分对 BYTE 分布 .....	2
Figure 2.	CSI-2 短包 .....	3
Figure 3.	CSI-2 长包 .....	3
Figure 4.	外部晶振作为参考时钟示意图 .....	5
Figure 5.	外部晶振作为参考时钟串行器工作示意图 .....	5

<b>Figure 6.</b>	<b>PCLK 作为参考时钟示意图</b> .....	<b>6</b>
<b>Figure 7.</b>	<b>PCLK 作为参考时钟串行器工作示意图</b> .....	<b>6</b>
<b>Figure 8.</b>	<b>CSI-2 接口串行器参考时钟来源示意图</b> .....	<b>7</b>
<b>Figure 9.</b>	<b>FPD-LINK III ADAS 解串器输出 CSI-2 流程</b> .....	<b>9</b>

## 1 车载摄像头数据传输常见接口

车载摄像头数据传输中常见接口有两种，DVP（digital video port）和 MIPI CSI-2 接口，因此 TI ADAS FPD-LINK III SERDES 也主要支持以上两种接口。在实际应用中，DVP 接口作为并行数据接口主要用于传输分辨率比较低的摄像头，比如 1 百万或者低于 1 百万摄像头；高分辨率摄像头像 2 百万，3 百万以及 8 百万等数据量比较大的摄像头主要使用 MIPI CSI-2 接口。

### 1.1 DVP 接口

DVP 接口作为并行数据接口，通常用在传输信号带宽不超过 100M 的情况，主要包括行同步信号（Hsync），帧同步信号（Vsync），时钟信号（PCLK）以及一组数据信号（Data），其中数据位宽有 8bit、10bit、12bit、16bit 等。TI 目前支持 DVP 接口的 ADAS FPD-LINK III 串行解串器最多支持 12bit 位宽，这代表着 1PCLK 内，串行器最多完成 12bits 数据采样，解串器最多完成 12bits 数据输出。

### 1.2 CSI-2 接口

CSI-2 接口作为标准协议接口，主要包括 1 个时钟差分对（Lane）以及 1-4 个数据差分对（Lane）。如图 1 所示，与 DVP 接口不同，CSI-2 基于字节（Byte）传输数据，每个字节支持短包（Short packet）或者长包（Long packet）格式，其中短包为固定长度 4bytes 数据包，主要用来传输帧开始/结束信号，行开始/结束信号，数据类型以及校验信息；而长包长度不定，主要用来传输数据本身，数据类型，校验信息等，总的来说 CSI-2 数据包中主要包括三点信息：

- 传输数据本身类型信息，帧、行同步信息
- 传输数据本身
- 校验信息

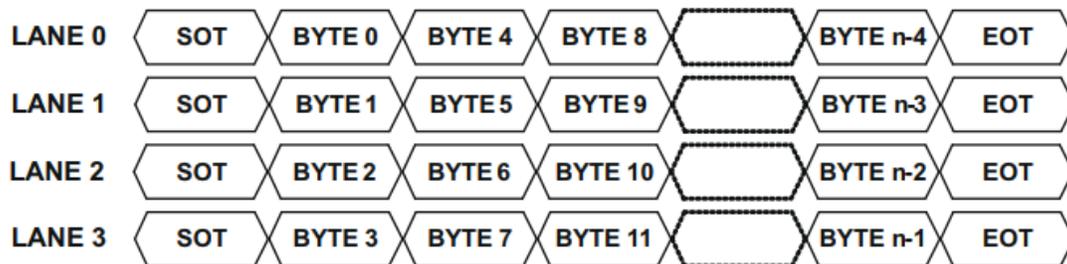


Figure 1. CSI-2 数据差分对 BYTE 分布

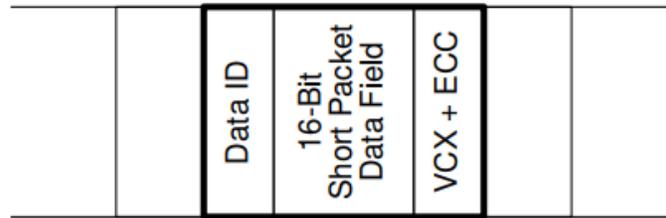


Figure 2. CSI-2 短包

短包架构:

- VC ID 以及数据类型 / 8bits
- 帧开始/结束或者行开始/结束 / 16bits
- 误差检测和校正 / 8bits

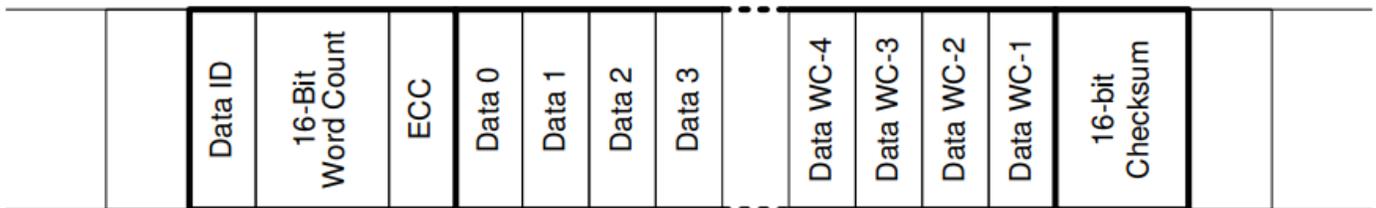


Figure 3. CSI-2 长包

长包架构:

- VC ID 以及数据类型 / 8bits
- World counts / 16bits
- ECC / 8bits
- 传输数据本身，比如 RAW10/12，YUV422 等视频数据
- CRC / 16bits

更多有关 CSI-2 数据传输应用层，协议层，物理层资料请参考 MIPI CSI-2 协议，本文不再赘述。

## 2 ADAS FPD-LINK III 串行器介绍

TI ADAS FPD-LINK III 串行器产品目前可选型号如下表所示。

Table 1. ADAS FPD-LINK III 串行器产品可选型号

接口类型	串化器可选型号	输入带宽	备注
DVP	DS90UB913A-Q1	25-100MHz PCLK	四者 P2P，四者主要区别在于输入带宽下限不同，另外 DS90UB633A-Q1 只能与 DS90UB6xx-Q1 解串器搭配。
	DS90UB913Q-Q1	10-100MHz PCLK	
	DS90UB933-Q1	37.5-100MHz PCLK	
	DS90UB633A-Q1	56.25-100MHz PCLK	

CSI-2	DS90UB935-Q1	4 data lanes, 80-832Mbps / Lane Total 2.528Gbps	四者 P2P，四者主要区别在于输入带宽上下限不同，DS90UB953A-Q1 相比于其他串行器支持环境温度范围更宽，为-40-125℃，其他为-40-100℃，另外 DS90UB635-Q1 只能与 DS90UB6xx-Q1 解串器搭配。
	DS90UB953-Q1	4 data lanes, 80-832Mbps / Lane Total 3.328Gbps	
	DS90UB953A-Q1	4 data lanes, 80-832Mbps / Lane Total 3.328Gbps	
	DS90UB635-Q1	4 data lanes, 600-832Mbps / Lane Total 3.328Gbps	

关于以上器件如何选型请参考参考文献 6，参考文献 6 详细介绍了不同应用，不同接口搭配下器件如何选型，本文不再赘述。

## 2.1 DVP 接口串行器工作原理

DS90UB933-Q1 作为目前车载 1 百万摄像头图像传输应用中常用器件之一，其可以用来传输像素时钟（PCLK）为 37.5-100MHz 的摄像头图像，本文接下来借助 DS90UB933-Q1 为例阐述 DVP 接口串行器基本工作原理。

DS90UB933-Q1 作为串化器主要是将 10/12-bit 并行数据结合双向控制信号串化成 FPD-LINK III 串行信号，并且通过线缆传输给解串器，解串器再将其解串成后端 SoC 所需要的接口信号，比如 DVP 或者 CSI-2 接口信号。DS90UB933-Q1 可以通过寄存器 0x05 选择工作模式以及 MODE 引脚选择其内部工作参考时钟来源，具体如下所示：

- 通过寄存器 0x05 配置工作模式
  - 10-bit 模式
  - 12-bit HF 模式
  - 12-bit LF 模式
- 通过 MODE 引脚选择参考时钟来源
  - 外部晶振
  - 图像传感器（Imager sensor）像素时钟

值得注意的是，上述内容提到的 10-bit 模式和 12-bit HF/LF 模式与图像编码格式 RAW10，RAW12 并无相关性，其仅仅指的是 DS90UB933-Q1 串化数据编码格式的不同，其中 10-bit 模式下，UB933 每两个 PCLK 将输入并行信号编码成 1 个 28bit 数据帧；12-bit LF 模式下，UB933 每个 PCLK 将输入并行信号编码成 1 个 28bit 数据帧；12-bit HF 模式下，UB933 每三个 PCLK 将输入信号编码成 2 个 28bit 数据帧。

除去编码格式不同之外，不同参考时钟源 DS90UB933-Q1 工作原理也不同，接下来以 10-bit 模式为例介绍 DS90UB933-Q1 不同参考时钟来源工作原理。

### 2.1.1 外部晶振作为参考时钟

如图 4 所示，当 DS90UB933-Q1 参考时钟为外部晶振时，其参考时钟经过内部锁相环（PLL）28 倍频将图像传感器输入的 10-bits 数据以及额外的 4-bits（双向控制信息，parity bit 和 CRC bits）串化成 FPD-LINK III 信号给到解串器，与此同时参考时钟也会经过 DS90UB933-Q1 串行器中的 2 分频电路输出给到图像传感器作为参考时钟。

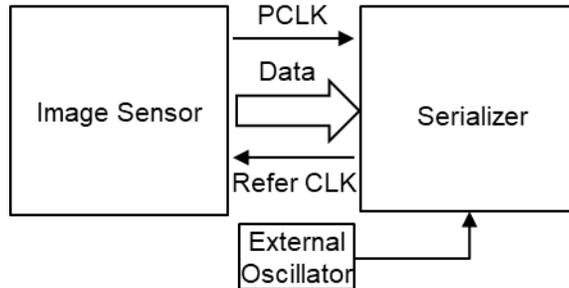


Figure 4. 外部晶振作为参考时钟示意图

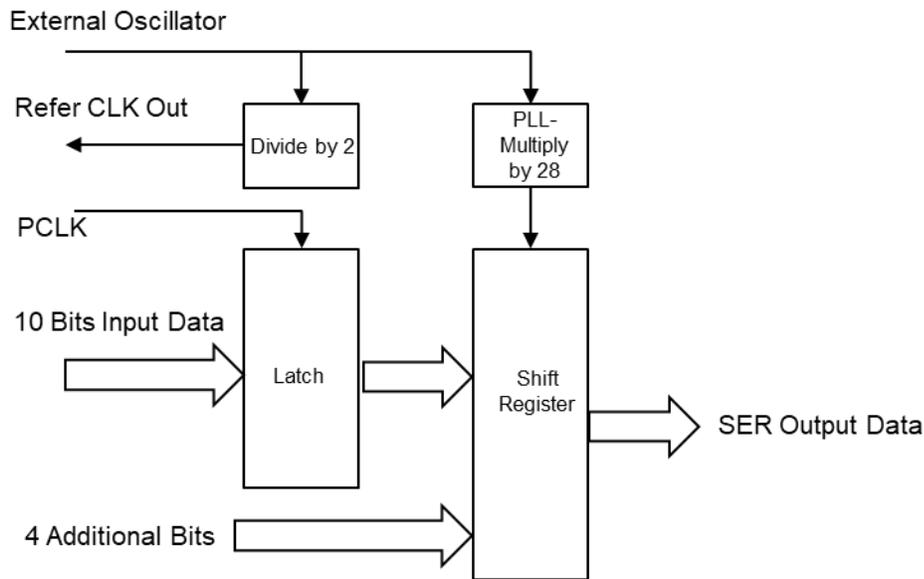


Figure 5. 外部晶振作为参考时钟串行器工作示意图

### 2.1.2 图像传感器输出 PCLK 作为参考时钟

当 DS90UB933-Q1 参考时钟为图像传感器像素时钟时，其参考时钟经过内部锁相环（PLL）14 倍频将图像传感器输入的 10-bits 数据以及额外的 4-bits（双向控制信息，parity bit 和 CRC bits）串化成 FPD-LINK III 信号给到解串器，此时图像传感器的参考时钟为外部晶振。

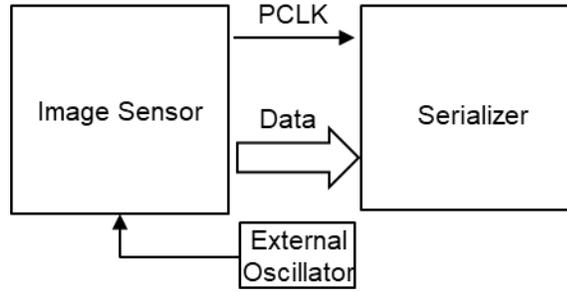


Figure 6. PCLK 作为参考时钟示意图

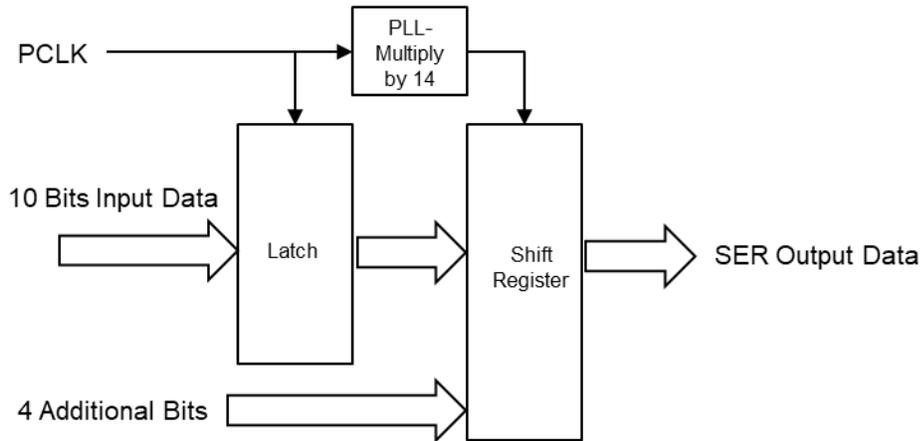


Figure 7. PCLK 作为参考时钟串行器工作示意图

### 2.1.3 两种参考时钟模式优劣势对比

当 DS90UB933-Q1 使用图像传感器输出 PCLK 作为参考时钟时，此时外部晶振直接给到图像传感器，外部晶振经过图像传感器内部 PLL 之后输出 PCLK 通常会增加一些抖动（Jitter），导致超出串行器规格（0.45UI），进而使得相对应的解串器不能正确解串出输入信号。

Table 2. DVP 接口串行器参考时钟模式优劣势对比

参考时钟模式	优势	劣势
外部晶振	参考时钟抖动小，生成 FPD-LINK 信号眼图眼宽宽，解串器不易无码。	如果输入并行信号 PCLK 不同，所需参考晶振不同，不灵活。
图像传感器 PCLK	参考时钟抖动大，生成 FPD-LINK 信号眼图眼宽不宽，解串器解析信号易误码。	如果输入并行信号 PCLK 不同时，所需参考晶振可以相同。

## 2.2 CSI-2 接口串行器工作原理

DS90UB935-Q1 作为目前车载 2 百万摄像头图像传输应用中常用器件之一，其可以用来接收 320M-2.528Gbps 带宽 CSI-2 信号，本文接下来以 DS90UB935-Q1 为例阐述 CSI-2 接口串行器基本工作原理。

DVP 接口串行器输出 FPD-LINK III 正向通道信号频率取决于串行器工作模式（10-bit 或者 12-bit）以及输入并行信号 PCLK 频率。相比于 DVP 接口串行器，CSI-2 接口串行器不再传输 CSI-2 信号时钟信号，其输出 FPD-LINK III 信号速率取决于串行器时钟模式，参考时钟信号频率以及 CLKIN\_DIV 配置，与输入 CSI-2 信号速率无关。

与 DVP 接口串行器一样，DS90UB935-Q1 同样地可以通过 MODE 引脚选择其工作模式以及参考时钟来源：

- CSI-2 同步模式，参考时钟来自解串器
- CSI-2 异步模式，参考时钟来自 CLKIN 引脚外部输入
- CSI-2 异步模式，参考时钟来自其内部 AON 时钟
- DVP 模式，参考时钟来自 CLKIN 引脚外部输入

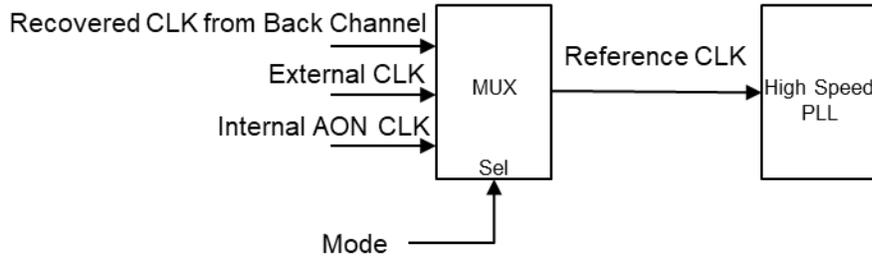


Figure 8. CSI-2 接口串行器参考时钟来源示意图

DS90UB935-Q1 等 CSI-2 接口串行器相比于 DVP 接口串行器增加了参考时钟来源于反向通道的功能，这也是其在实际应用中常用的时钟模式。在该模式下，DS90UB935-Q1 利用系统内部 CDR 模块恢复出反向通道数据和时钟信号，将该时钟信号作为 DS90UB935-Q1 串行输出正向通道信号参考时钟，并且此时 DS90UB935-Q1 也可经过内部 PLL 配置输出相应参考时钟给到图像传感器。因此在该时钟模式下，串行器系统（摄像头）中不再需要额外的外部晶振，不同时钟模式详细优劣势对比可见表 3。

Table 3. CSI-2 接口串行器参考时钟模式优劣势对比

时钟模式	优势	劣势
同步时钟模式	<ul style="list-style-type: none"> <li>• 摄像头系统中不需要额外晶振，进一步减小体积、成本和 EMC。</li> <li>• 接在同一解串器的多个摄像头参考时钟来自于同一时钟，不需要额外的同步。</li> <li>• 支持更快的反向通道速率。</li> </ul>	<ul style="list-style-type: none"> <li>• 接在同一解串器的不同摄像头图像传感器如果需要不同频率的参考时钟，存在配置不出特定频率时钟信号输出的情况。</li> </ul>

异步时钟模式	<ul style="list-style-type: none"> <li>接在同一解串器的不同摄像头图像传感器如果需要不同频率的参考时钟，可以通过配置串行器时钟分频参数获得，比较灵活。</li> </ul>	<ul style="list-style-type: none"> <li>摄像头系统中需要额外晶振，增加体积、成本和 EMC。</li> <li>接在同一解串器的多个摄像头参考时钟来自于同一时钟，需要额外的同步。</li> <li>反向通道速率较低，PoC 电感体积更大。</li> </ul>
--------	---	---

### 3 ADAS FPD-LINK III 解串器介绍

Table 4. ADAS FPD-LINK III 解串器产品可选型号

接口类型	解串器可选型号	输出 DVP 带宽		备注
DVP	DS90UB934-Q1	37.5-100MHz PCLK		三者 P2P，四者主要区别在于输出带宽下限不同。
	DS90UB914A-Q1	25-100MHz PCLK		
	DS90UB914Q-Q1	10-100MHz PCLK		
接口类型	解串器可选型号	输入 CSI-2 带宽	输出 CSI-2 带宽	备注
CSI-2	DS90UB638-Q1	1 port * 3.328Gbps	6.656Gbps	注意 DS90UB6xx-Q1 解串器只能与 DS90UB6xx-Q1 串行器搭配。
	DS90UB936-Q1	2 port * 2.528Gbps	6.656Gbps	
	DS90UB954-Q1	2 port * 3.328Gbps	6.656Gbps	
	DS90UB964-Q1	4 port * 1.330Gbps	12.8Gbps	
	DS90UB962-Q1	4 port * 2.528Gbps	6.656Gbps	
	DS90UB662-Q1	4 port * 2.528Gbps	6.656Gbps	
	DS90UB960-Q1	4 port * 3.328Gbps	13.312Gbps	

关于以上器件如何选型请参考参考文献 6，参考文献 6 详细介绍了不同应用，不同接口搭配下器件如何选型，本文不再赘述。

#### 3.1 DVP 接口解串器工作原理

由于 DVP 接口串行器 10-bit 模式和 12-bit HF/LF 模式编码格式不一致，所以 DVP 接口解串器例如 DS90UB934-Q1 同样可以通过 MODE 引脚或者寄存器 0x6D 选择工作在 RAW 10, RAW12 HF/LF 模式以及 FPD-LINK III 同轴线或者双绞线模式，以匹配前端串行器串化的 FPD-LINK III 数据。

相比于 CSI-2 接口串行解串器，DVP 接口串行器搭配 DVP 接口解串器工作原理较为简单，串行器将输入数据、行场同步信号、时钟信号串化成 FPD-LINK III 信号通过线缆传输到解串器，解串出数据、行场同步信号、时钟信号，串行器输入信号和解串器输出信号完全相同，唯一区别只在于少许延迟。

### 3.2 CSI-2 接口解串器工作原理

TI ADAS FPD-LINK III CSI-2 接口解串器主要分三种：单路、双路和四路 FPD-LINK III 输入解串器，其中单路、两路输入解串器通常用于 DMS、DVR 等应用，四路输入解串器通常用于环视等应用。图 8 以四路解串器为例介绍解串器将 FPD-LINK III 信号生成 CSI-2 信号流程：**FPD-LINK III 信号经过 AEQ 模块补偿因线缆传输而衰减的高频信号，以及时钟数据恢复模块（CDR）恢复时钟信号之后，最后解串得到的数据便会存放在每个 RX 缓存区域中，当 CSI 发生器检测到四个 RX 缓存区域中数据都准备妥当时，按照 CSI-2 寄存器相关配置开始生成 CSI-2 信号。其中 CSI-2 寄存器相关配置包括：**

- 0x33 / enable CSI-2 port 0 or 1
- 0x20 / map each RX port to each CSI-2 port
- 0x31 / select CSI-2 forwarding mode

CSI-2 forwarding mode 分为四种模式：Round-Robin; Basic synchronized CSI-2 forwarding; Line-Interleave forwarding; Line-Concatenated forwarding，其中后三种模式也统称为同步模式，其中 Round-Robin 模式对每个 RX port 的数据类型，分辨率，每行数据到达时间无特别要求，但输出 CSI-2 信号具有 4 个 VC ID；而同步模式需要每个 RX port 的数据类型，分辨率保持相同并且每行数据到达时间误差在一行以内，输出 CSI-2 信号 VC ID 根据不同配置可为 4 个（Basic synchronized CSI-2 forwarding）或者 1 个（Line-Interleave & Concatenated forwarding），更加详细对比参考参考文献 7。

因此对于每路输入分辨率不同，串化器不同的应用，只能使用 Round-Robin 模式，而对于环视应用，四种模式皆可使用，但最常用的是 Line-Interleave forwarding 和 Line-Concatenated forwarding 模式，主要是因为这两种模式输出 CSI-2 信号中 VC ID 相同，可以减少后端 SoC VC ID 资源利用。同步模式中解串器会通过反向通道分别给四个摄像头图像传感器（Image Sensor）提供帧同步信号（Frame sync signal），图像传感器收到帧同步信号之后则会开始输出图像数据给到串化器，从而使得 4 个摄像头图像数据在一行误差内传输到解串器。

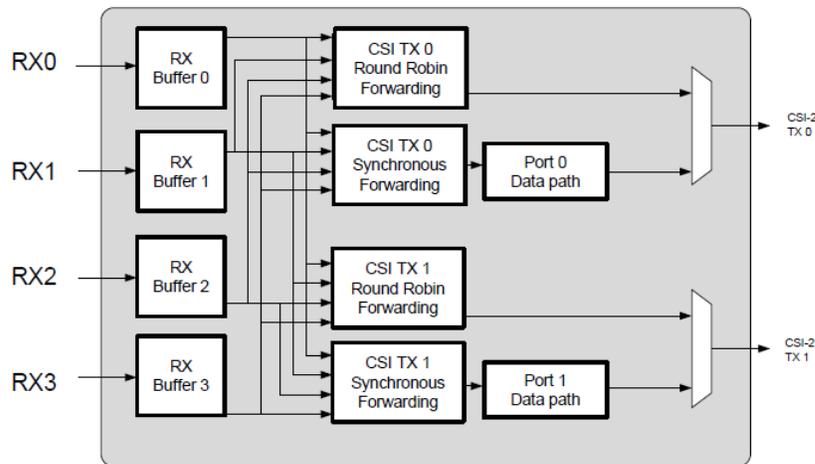


Figure 9. FPD-LINK III ADAS 解串器输出 CSI-2 流程

## 4 ADAS FPD-LINK III 示例代码

为加快用户调试过程，本节给出了 ADAS FPD-LINK III 不同接口搭配场景示例代码，用户可根据实际需求进行相应修改。

### 4.1 DVP 接口串行器搭配 DVP 接口解串器

该应用场景主要用于单颗 1 百万摄像头传输，针对此应用场景用户在硬件配置完之后不需要做任何软件配置，串行器解串器便能 lock，并且解串器会自动在上电时根据 Mode 引脚配置串行器工作模式（10bit 或者 12bit）。

### 4.2 DVP 接口串行器搭配 CSI-2 接口解串器

该应用场景主要用在 1 百万摄像头环视应用，本文以 4\*DS90UB933-Q1 搭配 DS90UB964-Q1 为例，示例代码如下。

```
board.WriteI2C(TI964, 0x01, 0x02); //reset all digital part
time.sleep(0.5)

## Internally generated framesync configuration
board.WriteI2C(TI964, 0x0C, 0x0F); //enable RX0-3
board.WriteI2C(TI964, 0x4C, 0x0F); //write enable for RX0-4 registers
board.WriteI2C(TI964, 0x4C, 0x01); //page to port RX0
board.WriteI2C(TI964, 0x6E, 0xAA); //set RX0 SER GPIO1 output framesync signal
board.WriteI2C(TI964, 0x4C, 0x12); //page to port RX1
board.WriteI2C(TI964, 0x6E, 0xAA); //set RX1 SER GPIO1 output framesync signal
board.WriteI2C(TI964, 0x4C, 0x24); //page to port RX2
board.WriteI2C(TI964, 0x6E, 0xAA); //set RX2 SER GPIO1 output framesync signal
board.WriteI2C(TI964, 0x4C, 0x38); //page to port RX3
board.WriteI2C(TI964, 0x6E, 0xAA); //set RX3 SER GPIO1 output framesync signal
board.WriteI2C(TI964, 0x19, 0x01); //FS_HIGH_TIME_1
board.WriteI2C(TI964, 0x1A, 0x15); //FS_HIGH_TIME_0
board.WriteI2C(TI964, 0x1B, 0x09); //FS_LOW_TIME_1
board.WriteI2C(TI964, 0x1C, 0xC3); //FS_LOW_TIME_0
board.WriteI2C(TI964, 0x18, 0x01); //enable framesync

## RX0-3 I2C pass through and slave ID configuration
board.WriteI2C(TI964, 0x4C, 0x01); //page to port RX0
board.WriteI2C(TI964, 0x58, 0x58); //enable passthrough
board.WriteI2C(TI964, 0x5D, snrID); //imager slave ID, port0
board.WriteI2C(TI964, 0x65, aliasID0); //imager slave alias, port0
board.WriteI2C(TI964, 0x7C, 0x01); //set frame valid polarity to active high

board.WriteI2C(TI964, 0x4C, 0x12); //page to port RX1
board.WriteI2C(TI964, 0x58, 0x58); //enable passthrough
board.WriteI2C(TI964, 0x5D, snrID); //imager slave ID, port1
board.WriteI2C(TI964, 0x65, aliasID1); //imager slave alias, port1
board.WriteI2C(TI964, 0x7C, 0x01); //set frame valid polarity to active high

board.WriteI2C(TI964, 0x4C, 0x24); //page to port RX2
board.WriteI2C(TI964, 0x58, 0x58); //enable passthrough
```

```

board.WriteI2C(TI964, 0x5D, snrID); //imager slave ID, port2
board.WriteI2C(TI964, 0x65, aliasID2); //imager slave alias, port2
board.WriteI2C(TI964, 0x7C, 0x01); //set frame valid polarity to active high

board.WriteI2C(TI964, 0x4C, 0x38); //page to port RX3
board.WriteI2C(TI964, 0x58, 0x58); //enable passthrough
board.WriteI2C(TI964, 0x5D, snrID); //imager slave ID, port3
board.WriteI2C(TI964, 0x65, aliasID3); //imager slave alias, port3
board.WriteI2C(TI964, 0x7C, 0x01); //set frame valid polarity to active high

## CSI output configuration for basic synchronized forwarding
board.WriteI2C(TI964, 0x4C, 0x01); //page to port RX0
board.WriteI2C(TI964, 0x70, 0x1F); //RAW10_datatype_yuv422b10_VC0
board.WriteI2C(TI964, 0x7C, 0xC0); //RAW10 use lower 8bit
board.WriteI2C(TI964, 0x4C, 0x12); //page to port RX1
board.WriteI2C(TI964, 0x70, 0x5F); //RAW10_datatype_yuv422b10_VC1
board.WriteI2C(TI964, 0x7C, 0xC0); //RAW10 use lower 8bit
board.WriteI2C(TI964, 0x4C, 0x24); //page to port RX2
board.WriteI2C(TI964, 0x70, 0x9F); //RAW10_datatype_yuv422b10_VC2
board.WriteI2C(TI964, 0x7C, 0xC0); //RAW10 use lower 8bit
board.WriteI2C(TI964, 0x4C, 0x38); //page to port RX3
board.WriteI2C(TI964, 0x70, 0xDF); //RAW10_datatype_yuv422b10_VC3
board.WriteI2C(TI964, 0x7C, 0xC0); //RAW10 use lower 8bit
board.WriteI2C(TI964, 0x32, 0x01); //read from CSI port0, write to CSI port0
board.WriteI2C(TI964, 0x33, 0x03); //enable CSI port0
board.WriteI2C(TI964, 0x32, 0x12); //read from CSI port1, write to CSI port1
board.WriteI2C(TI964, 0x33, 0x03); //enable CSI port1
board.WriteI2C(TI964, 0x20, 0x0C); //[3:0]: 0: CSI-2 port 0; 1: CSI-2 port 1;
board.WriteI2C(TI964, 0x21, 0x14); //enable basic Synchronized forwarding for CSI port 0&1

```

### 4.3 CSI-2 接口串行器搭配 CSI-2 接口解串器

该应用场景主要用于单颗 2 百万摄像头传输或者 2 百万摄像头环视应用，本文以 4\*DS90UB935-Q1 搭配 DS90UB960-Q1 为例，示例代码如下（单颗 2 百万摄像头只需配置单 Port 即可）。

```

board.WriteI2C(TI960, 0x01, 0x02); //reset all digital part
time.sleep(0.5)

```

```

## Internally generated framesync configuration

```

```

## RX0-3 I2C pass through and slave ID configuration

```

上述代码与 DVP 接口串行器搭配 CSI-2 接口解串器示例代码一致，唯一不同在于 CSI 输出配置代码。

```

## CSI output configuration for basic synchronized forwarding
board.WriteI2C(TI960, 0x4C, 0x01); //page to port RX0
board.WriteI2C(TI960, 0x4C, 0x12); //page to port RX1
board.WriteI2C(TI960, 0x4C, 0x24); //page to port RX2
board.WriteI2C(TI960, 0x4C, 0x38); //page to port RX3
board.WriteI2C(TI960, 0x32, 0x01); //read from CSI port0, write to CSI port0
board.WriteI2C(TI960, 0x33, 0x03); //enable CSI port0
board.WriteI2C(TI960, 0x32, 0x12); //read from CSI port1, write to CSI port1
board.WriteI2C(TI960, 0x33, 0x03); //enable CSI port1

```

```
board.WriteI2C(TI960, 0x20, 0x0C); //[3:0]: 0: CSI-2 port 0; 1: CSI-2 port 1;  
board.WriteI2C(TI960, 0x21, 0x03); //enable round robin forwarding for CSI port 0&1
```

#### 4.4 CSI-2 接口串行器搭配 DVP 接口解串器

DVP 接口由于作为并行数据接口，通常所传输数据量小于 CSI-2 接口，因此如果使用 DVP 接口解串器通常会搭配 DVP 接口串行器，这种组合的应用场景比较少。此种搭配使用，可参考参考文献 7，此处不再展开。

## 5 参考文献

1. *DS90UB933-Q1 datasheet (SNLS546E)*
2. *DS90UB934-Q1 datasheet (SNLS507C)*
3. *DS90UB935-Q1 datasheet (SNLS605C)*
4. *DS90UB964-Q1 datasheet (SNLS500A)*
5. *DS90UB960-Q1 datasheet (SNLS589C)*
6. *深入理解 FPD-LINK 产品的速率评估方法 (ZHCAA76)*
7. *深入理解 FPD-LINK III ADAS 解串器 HUB 产品 (ZHCACI8)*
8. *Backwards Compatibility Modes for Operation With Parallel Output Deserializers (SNLA270A)*

## 重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2024，德州仪器 (TI) 公司