Line buffer architecture spec

**ARCHITECTURE**

Line buffer结构和连接关系如下所示：





**APPLICATION SCENARIO**

**Write Operation**

HDMI和MIPI不会同时进行写操作；

DVP，HDMI和MIPI不会同时进行写操作；**(**待确定**)**

**Read Operation**

**Case1**: ISP work only;

**Case2**: BYPASS work only;<分辨率的范围？？up to 8K>

**Case3**: ISP& BYPASS work simultaneously;<最大分辨率为4K??>

Case3.1: ISP&BYPASS read one same picture;

**RAM\_SIZE:**

与双摄像头之间的延时及总线响应BYPASS模块的延时有关，便于测试，目前暂定为1.5line;  <TBD>

RAM\_0: 1440X84;

RAM\_1: 1440X84;

RAM\_2: 1440X84;

RAM\_3: 1440X84;

RAM\_4: 1440X84;

RAM\_5: 1440X84;

**读写控制：**

双目视觉及4K传输需要相关RAM空间合并，由系统寄存器控制，相关寄存器控制如下：<寄存器列表见最后>

**SCENARIO\_SELCET**: 1 four\_lane+ 4two\_lane; 6 two\_lane;

**VIDEO\_SOURCE\_RAM01[2:0]**: bit[0], bit[1],bit[2] for DVP,HDMI,MIPI

**VIDEO\_MERGE\_ISP\_0[5:0]：**

**VIDEO\_MERGE\_ISP\_1[5:0]；**

**VIDEO\_MERGE\_ISP\_2[5:0]；**

**VIDEO\_MERGE\_ISP\_[x][5:3]& VIDEO\_MERGE\_ISP\_[x][2:0]分别代表左右摄像头；**

**VIDEO\_MERGE\_ISP\_[x][5:3]== VIDEO\_MERGE\_ISP\_[x][2:0]只从单摄像头取数据；**

**WR\_POINTER\_THEN\_READ\_PATH\_[x][10:0] RAM[X]开始读数的节点；**

**INITI\_FRAME\_TIME[3:0]： camera初始化的时间设置；**

**CLOCK\_SYNC\_THRESHOLD[9:0]：双目视觉时两个camera时钟同步的阈值；**

**ISP接口**

ISP的处理能力1 pixel/cycle, Liner buffer以发图的方式给ISP传送图像，接口时序如下所示：



**DIRECTION**: 双目时指示前后半行；0表示左半部分，1表示右半部分；默认值为0。

**V\_SYNC**: 帧同步信号；脉冲信号

**H\_SYNC**: 行传输指示信号；高有效；

**PIX\_EN**: 数据有效信号；高有效；

**ODD\_LINE:** 偶数行，YUV\_format下有效，高有效，默认值为0;

**ODD\_PIXEL:** YUV\_format下有效，pixel处在偶数位置，高有效，默认值0；

**PIX\_DATA**[29:0]: 最差情况为YUV-10bit; 数据Mapping方式:

RGB/YUV: 三分量分别对应[29:20]、[19:10]、[9:0]；

RAW : [14:0] ；低位对齐；

SWITCH\_DELAY的要求？？？

**BYPSS接口**

ISP与BYPASS同时工作时，ISP的优先级高。

BYPASS模块接口与ISP接口时序相似，如下所示



**Rd\_en:** BYPASS反馈的读使能信号，高有效；

**V\_SYNC**: 帧同步信号；

**H\_SYNC**: 行传输指示信号；高有效；

**Data\_type**: 数据类型；

**PIX\_EN**: 数据有效信号；高有效；

**ODD\_LINE:** 偶数行，YUV\_format下有效，高有效，默认值为0;

**ODD\_PIXEL:** YUV\_format下有效，pixel处在偶数位置，高有效，默认值0；

**PIX\_DATA**[29:0]: 最差情况为YUV-10bit;

数据Mapping:

RGB/YUV: 三分量分别对应[29:20]、[19:10]、[9:0]；

RAW : [14:0] ；低位对齐；

**APB\_Config**

利用ISP apb接口；？？？

需要系统配置的参数如下：

|  |  |
| --- | --- |
| **Register** | **Description** |
| SCENARIO\_SELCET | 0：6 2lane\_mipi;  1: 1 4lane\_mipi+4 2lane\_mipi; |
| FRAME\_LENGTH\_PATH\_0[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_0[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_0[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_0[9:0] | The value of write pointer When begin to read |
| FRAME\_LENGTH\_PATH\_11[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_1[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_1[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_1[9:0] | The value of write pointer When begin to read |
| FRAME\_LENGTH\_PATH\_2[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_2[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_2[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_2[9:0] | The value of write pointer When begin to read |
| FRAME\_LENGTH\_PATH\_3[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_3[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_3[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_3[9:0] | The value of write pointer When begin to read |
| FRAME\_LENGTH\_PATH\_4[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_4[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_4[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_4[9:0] | The value of write pointer When begin to read |
| FRAME\_LENGTH\_PATH\_5[12:0] | The pixel number of one line |
| FRAME\_WIDTH\_PATH\_5[12:0] | The number of line in one frame |
| DATA\_TYPE\_PATH\_5[5:0] | The data type of the frame data |
| WR\_POINTER\_THEN\_READ\_PATH\_5[9:0] | The value of write pointer When begin to read |
| INITI\_TIME[3:0] | Timing for camera initial, unit (frame) |
| CLOCK\_SYNC\_THRESHOLD[9:0] | Time to sync the two camera clock; unit (pixclk) |
| VIDEO\_SOURCE\_RAM01[2:0] | 0:DVP  1:HDMI  2:MIPI |
| VIDEO\_MERGE\_ISP\_0[5:0] | [5:3] Read first&[2:0] read later  If equal, read from one memory only |
| VIDEO\_MERGE\_ISP\_1[5:0] | [5:3] Read first&[2:0] read later  If equal, read from one memory only |
| VIDEO\_MERGE\_ISP\_2[5:0] | [5:3] Read first&[2:0] read later  If equal, read from one memory only |