



FPGA OS_5in_5out_mac 使用文档

主题	FPGA OS_5in_5out_mac 使用文档
文档号	
创建时间	2019-11-07
最后修改	2019-11-07
版本号	1.1
文件名	FPGA OS_5in_5out_mac 使用文档.pdf
文件格式	Portable Document Format



目录

一、 FAST 结构介绍.....	4
二、 FPGA OS 与 UM 接口定义.....	4
三、 数据分组结构定义.....	7
四、 接口分组定义.....	8
五、 数据报文 Metadata 格式定义.....	9
六、 设备 MAC 地址配置.....	9



修改记录

版本号	修改人	日期	备注
1.0	张彦龙	2019-08-09	初始版本
1.1	王耀祥	2019-11-07	添加设备 mac 地址 寄存器





一、FAST 结构介绍

FAST 结构如图 1 所示。其分为 FPGA_OS 和 UM 两部分。FPGA OS 包括 FPGA OS Ingress、FPGA OS egress 及 FPGA OS CDC 三部分；UM 由用户自己定义。

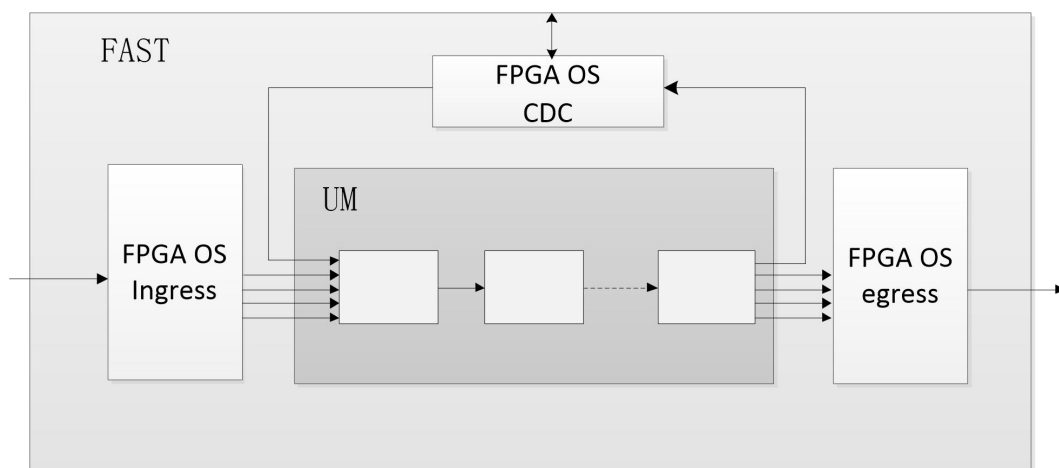


图 1 FAST 平台结构图

FPGA OS Ingress 部分：主要用于完成 1) 接口及 DMA 数据输入的汇聚及输入调度功能；2) 对接口输入的分组数据封装 FAST 的 Metadata 头。

FPGA OS Egress 部分：主要用于完成 1) 接口数据输出控制 2) 对接口输出的分组数据解封装 FAST 的 Metadata 头。

FPGA OS CDC 部分：主要用于完成 CPU 与硬件模块的数据交互。

二、FPGA OS 与 UM 接口定义

FPGA OS 内与 UM 的连接信号图如 2 所示。

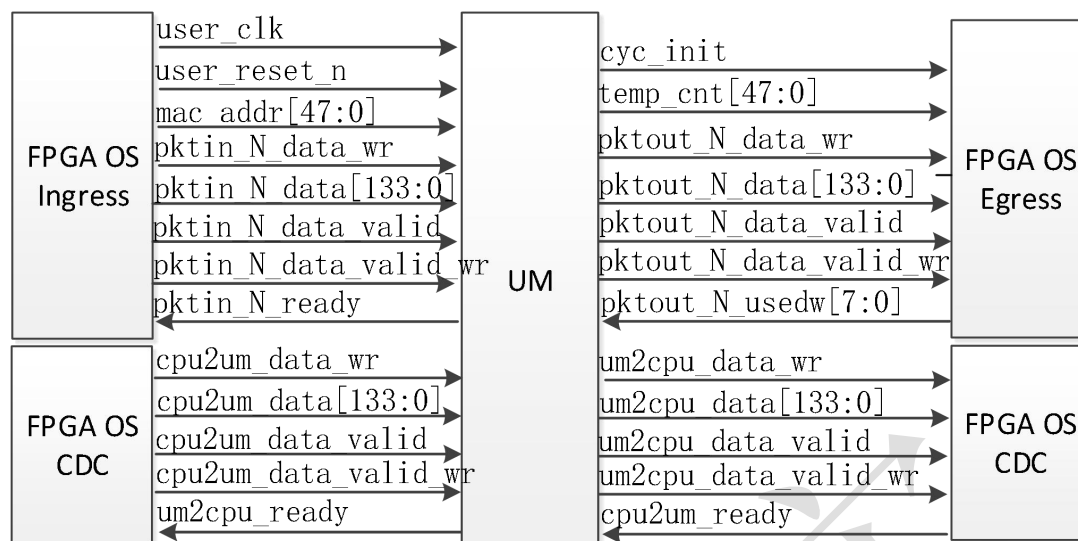


图 2 FPGA 与 UM 接口定义

接口信号定义及列表

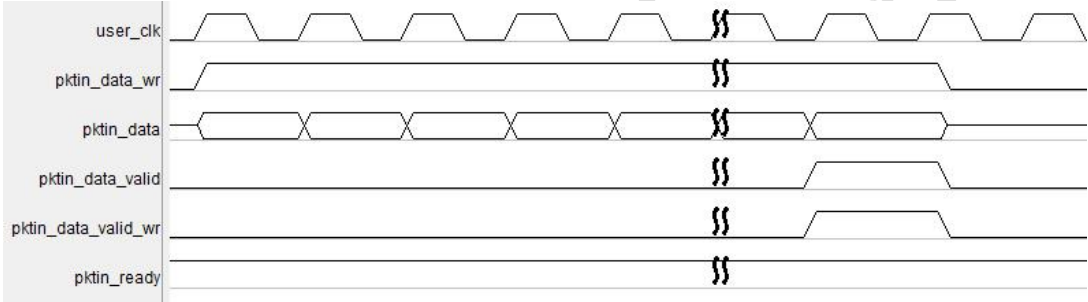
信号名	方向	位宽	描述
CLK and RESET			
user_clk	Input	1	125Mhz 的输入时钟
user_reset_n	Input	1	复位信号，低有效
FPGA OS Ingress to UM 信号定义			
pktin_N_data_wr	Input	1	报文数据写信号，N 为 0-3
pktin_N_data	Input	134	报文数据，N 为 0-3
pktin_N_data_valid	Input	1	报文数据标志位,1 为有效分组，0 为无效分组，N 为 0-3
pktin_N_data_valid_wr	Input	1	报文数据标志位写信号，N 为 0-3
pktin_N_ready	output	1	数据 ready 信号，N 为 0-3
mac_addr	input	48	设备本地 MAC 地址
FPGA OS CDC to UM 信号定义			
cpu2um_data_wr	Input	1	报文数据写信号
cpu2um_data	Input	134	报文数据
cpu2um_data_valid	Input	1	报文数据标志位,1 为有效分组，0 为无效分组。
cpu2um_data_valid_wr	Input	1	报文数据标志位写信号
um2cpu_ready	output	1	数据 ready 信号
UM to FPGA OS Egress 信号定义			
pktout_N_data_wr	output	1	输出报文写信号，N 为 0-3
pktout_N_data	output	134	输出报文数据，N 为 0-3
pktout_N_data_valid	output	1	输出报文标志位，N 为 0-3
pktout_N_data_valid_wr	output	1	输出 FIFO usedw 值，N 为 0-3
pktout_N_usedw	input	8	输出报文 ready 信号，N 为 0-3
UM to FPGA OS CDC 信号定义			



um2cpu_data_wr	output	1	报文数据写信号，N 为 0-3
um2cpu_data	output	134	报文数据
um2cpu_data_valid	output	1	报文标志位
um2cpu_data_valid_wr	output	1	报文标志位写信号
um2cpu_ready	input	1	报文 ready 信号
UM to FPGA OS 信号定义			
cyc_init	output	1	用于根据调整后的计数器值同步外围时钟计数的有效信号
temp_cnt	output	48	用于根据调整后的计数器值同步外围时钟计数，[47:17]单位为 ms，[16:0]单位为 8ns

接口时序

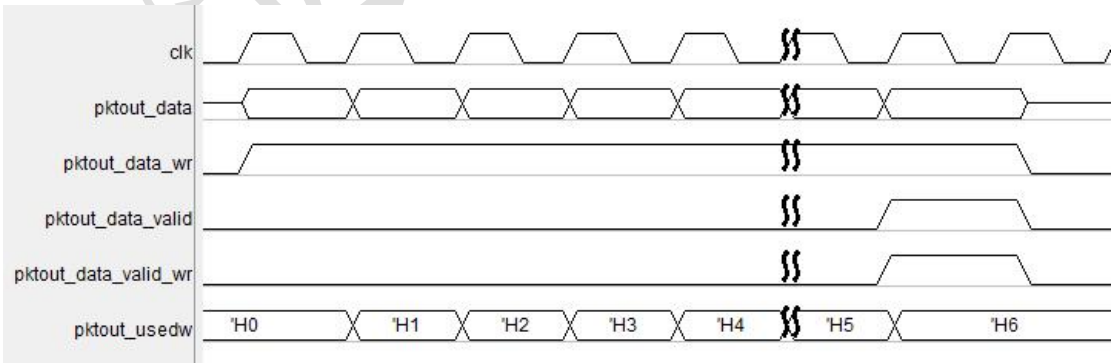
FPGA OS Ingress to UM 信号时序定义：



操作步骤：

- 1) 在 pktin_ready 输出有效时，即为 1 时，检测接收分组；
- 2) 当检测到 pktin_data_wr 为 1 时，表示数据分组 pktin_data 输入数据有效；
- 3) 当数据输入到最后一拍时，则 pktin_data_valid_wr 为 1，输入分组有效时 pktin_data_valid 为 1，否则为 0。

UM to FPGA OS Egress 信号时序定义：



操作步骤：

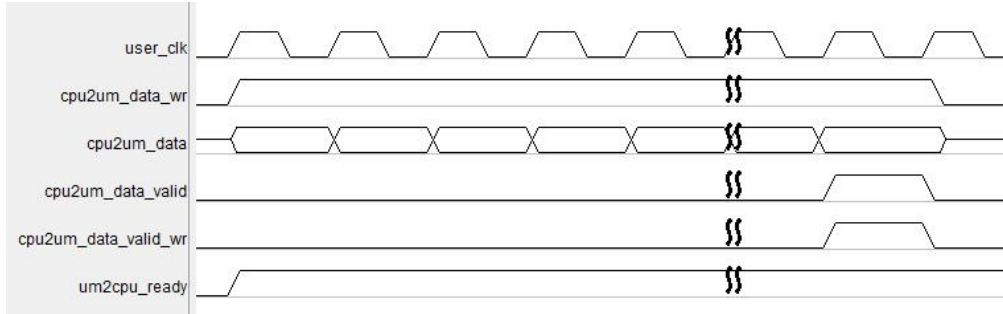
- 1) 在 pktout_usedw 为小于 8' d161 时，即可以缓存一个最长分组，输出分组；
- 2) 数据分组输出时即 pktout_data 输出数据时，置 pktout_data_wr 信号为



1;

- 3) 当数据输出到最后一拍时, 则 pktout_data_valid 与 pktout_data_valid_wr 为 1。

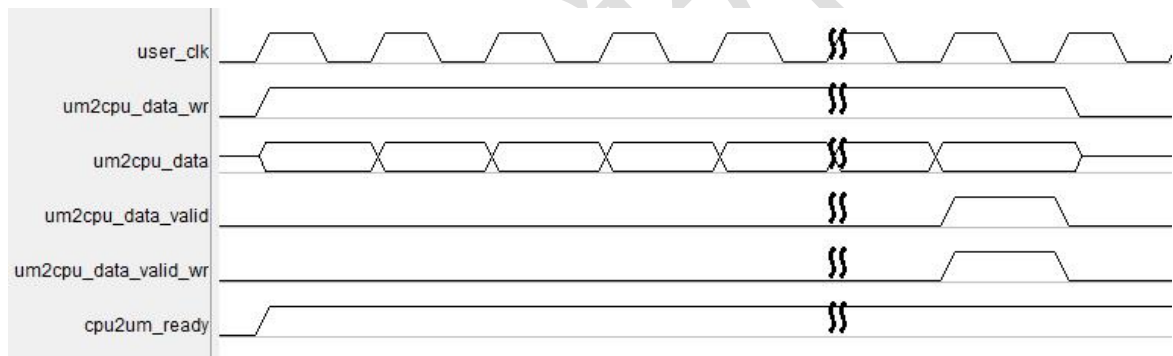
FPGA OS CDC to UM 信号时序定义:



操作步骤:

- 4) 在 um2cpu_ready 输出有效时, 即为 1 时, 检测接收分组;
5) 当检测到 cpu2um_data_wr 为 1 时, 表示数据分组 cpu2um_data 输入数据有效;
6) 当数据输入到最后一拍时, 则 cpu2um_data_valid_wr 为 1, 输入分组有效时 cpu2um_data_valid 为 1, 否则为 0。

UM to FPGA OS CDC 信号时序定义:



操作步骤:

- 1) 在 cpu2um_ready 输入有效时, 即为 1 时, 输出分组;
2) 数据分组输出时即 um2cpu_data 输出数据时, 置 um2cpu_data_wr 信号为 1;
3) 当数据输出到最后一拍时, 则 um2cpu_data_valid 与 um2cpu_data_valid_wr 为 1。

三、数据分组结构定义

输入及输出数据分组包括 Metadata 头部及有效数据分组两部分, 格式如图 3 所示, Metadata 在 FAST 报文的前 32 字节携带, 每个分组进出 UM 的第 1 拍 16 字节为 Metadata0, 第二拍数据为 Metadata1。

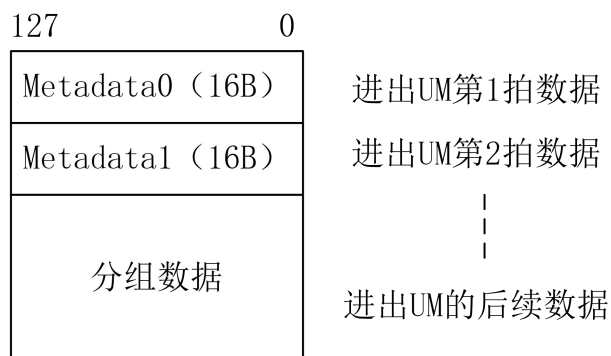


图 3 分组数据传输格式

四、 接口分组定义

接口分组(packet)是应用在 FPGA OS 与 UM 接口上的 134bit 的数据格式，其中高 6 位为控制信息，低 128 位为报文数据。分组的前两拍为 FPGA OS 添加的 32 字节的 metadata，两拍后的数据为有效分组数据。134 位的数据由 2 位的头尾标识，4 位无效字节数，128 位的有效数据组成。

其中，[133:132]位为报文数据的头尾标识，01 代表报文头部，11 代表报文中间数据，10 代表报文尾部；[131:128]位为 4 位的无效字节数，其中 0000 表示 16 个字节全部有效，0001 表示最低一个字节无效，最高 15 个字节有效，依次类推，1111 表示最低 15 个字节无效，最高一个字节有效。格式如图 4 所示。



图 4 报文分组传输格式



五、 数据报文 Metadata 格式定义

Metadata0 格式定义如下：

[127]	1	pktsrc	分组的来源，0 为网络接口输入，1 为 CPU 输入
[126]	1	pktdst	分组目的，0 为网络接口输出，1 为送 CPU
[125:120]	6	inport	分组的输入端口号
[119:118]	2	outtype	00:直接输出，01：查组播，10，11 保留
[117:112]	6	outport	直接输出：bitmap 方式输出，查组播表：为查找索引。
[111:109]	3	priority	分组优先级
[108]	1	discard	丢弃位
[107:96]	12	len	包含 Metadata 字段的分组长度
[95:88]	8	smid	最近一次处理分组的模块 ID
[87:80]	1	dmid	下一个处理分组的模块 ID
[79:72]	8	pst	标准协议类型，图 4 所示
[71:64]	8	seq	分组接收序列号
[63:50]	14	flowid	流 ID
[49:48]	17	reserve	保留
[47:0]	48	ts	时间戳

Metadata1:为用户预留的 16B 的自定义空间，用户可以根据自己需求，自定义内容及使用。

六、 设备 MAC 地址配置

在硬件逻辑中添加了一个 48 位的 mac 地址寄存器，用于配置设备本地 mac 地址，具体 MAC 地址如何配置请先参考软件说明。在硬件中信号已经引入到 UM 模块顶层，用户可在 um 中声明调用该信号，verilog 声明方法如下：

```
input wire [47:0] mac_addr
```