
HUNDSUN®

FPGA 极速行情服务系 V1.0

用户手册

恒生电子股份有限公司 公司研发中心

版权所有，侵权必究

文档版本 1.0.1

发布日期 2020-8-31

恒生电子股份有限公司 研发中心

地址： 杭州市滨江区江南大道 3588 号恒生大厦 14 楼 邮编：
310053

网址： <http://www.hundsun.com>

客户服务电话： 0571-28829563

客户服务传真： 0571-28823456

客户服务邮箱： service.rd@hundsun.com



版权所有 © 恒生电子股份有限公司 2020。 保留一切权利。


非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

密级： 内部

第 2 页，共 42 页

恒生电子

商标声明

 天堂恒生[®] 你我他[®] 为恒生电子股份有限公司的注册商标。

注意：

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

文档修改记录

版本	修订人	修订说明	批准人	发布日期
0.000	吴凤雏	初版发布		20210516
0.001	吴凤雏	1. 增加 3.3 中，升级功能说明 2. 修改加载驱动说明		20210707
0.002	吴凤雏、 马树	1. 修改 典型部署场景 2. 增加中断绑核说明		20210727
0.003	吴凤雏	1. 更新使用说明		20210911
0.004	吴凤雏	1. 增加上交所 TCP 行情版本说明		20210926
0.005	魏帅	1. 增加深交所 UDP 行情版本说明		20211221
0.006	魏帅	1. 增加 U50 板卡支持		20221221
0.007	马树	1 修改 SEE 部署方法：需要先选择节点 2 部署过程描述修改		20230317
0.008	马树	增加 2.1 节，描述驱动和固件知识		20230429
0.009	方建正	补充板卡安装说明		20230615
0.010	马树	SEE 部署说明修改		20230821
0.011	周振磊	补充不同 MPO-LC 线序接入方式		20240729

目录

一、 概述.....	8
1 背景.....	8
2 系统架构.....	8
2.1 网络版行情架构.....	8
2.2 本地版行情架构.....	9
3 FPGA 行情版本说明	9
二、 部署方案	10
1 典型的部署场景	10
1.1 TCP 网络版.....	10
1.2 UDP 网络版.....	10
1.3 UDP 本地版.....	11
2 部署要求.....	12
2.1 操作系统要求	12
2.2 服务器硬件要求.....	12
2.3 网络环境要求	12
三、 操作说明	12
1 板卡安装.....	12
1.1 板卡介绍.....	12
1.2 安装准备.....	14
2 驱动安装.....	18
2.1 驱动安装和固件关系概述	18
2.1.1 升级固件前提	18
2.1.2 查看当前板卡固件类型	18
2.1.3 驱动名称	18
2.2 xdma 驱动安装.....	19
2.2.1 脚本方式安装	19

2.2.2	手动命令安装	19
2.3	E 驱动安装	23
2.3.1	安装方法	23
2.3.2	部分安装错误以及解决方法	23
3	监控软件	24
3.1	监控软件安装	24
3.2	监控软件使用	24
3.2.1	监控程序的行为	25
3.2.2	监控项说明	25
3.3	配置文件说明	26
4	配置工具安装	28
4.1	配置工具安装	28
4.2	配置工具使用说明	28
4.3	配置文件说明	32
5	HPT 高精度时钟卡操作说明	32
5.1	HPT 卡的两种应用场景	32
5.1.1	同源对比	32
5.1.2	穿透对比	33
6	SEE 部署 FPGA 监控说明	33
四.	FPGA 深交所 TCP 网关 L2 行情网络版	34
1	支持业务功能范围	34
五.	FPGA 深交所 TCP 网关 L1 行情网络版	35
1	支持业务功能范围	35
六.	FPGA 上交所 TCP 网关前/网关后 L2 行情网络版/本地版	35
1	支持业务功能范围	35
七.	FPGA 上交所 TCP 网关 L1 行情网络版	36

1	支持业务功能范围	36
八.	FPGA 深交所 UDP 网关前/网关后 L2 行情网络版/本地版.....	36
1	支持业务功能范围	37
九.	FPGA 深交所 UDP 网关 L1 行情网络版/本地版	37
1	支持业务功能范围	37
十.	常见问题解决	37
1	SDK 收不到包的问题排查	37
2	如何确认 FPGA 正在正常收包.....	39
3	如何确认 FPGA 正在发包	41

一. 概述

1 背景

本用户手册介绍 FPGA 极速行情服务系统 V1.0 的部署、功能、模块和运行步骤等信息。

2 系统架构

行情系统从系统架构上讲，分成网络版和本地版。其中网络版提供一对多的服务，行情系统将解码后数据通过组播方式发送给多个 SDK；本地版与 SDK 部署在同一台服务器，只服务于本机的客户。

2.1 网络版行情架构

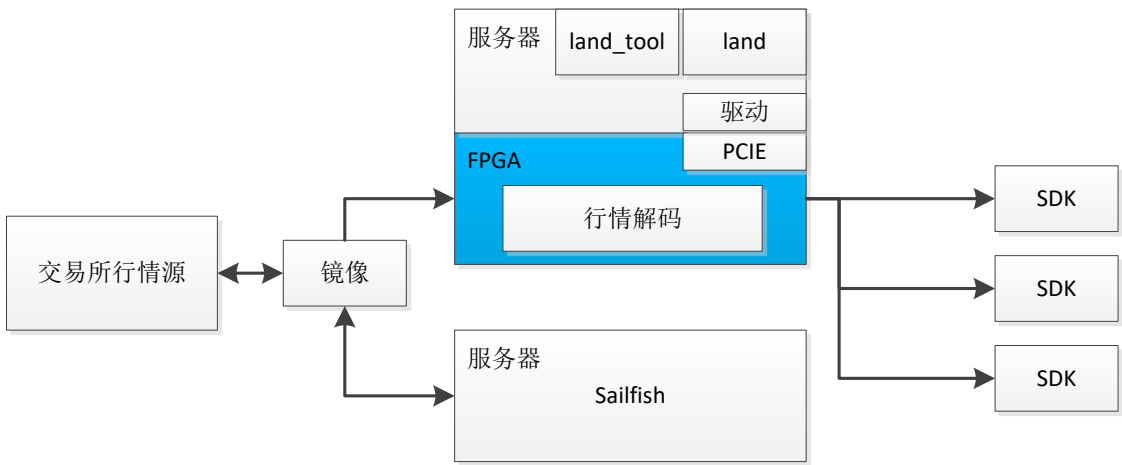


表 1.2.1 产品模块介绍

节点	说明
交易所行情源	MDGW/VDE 等行情网关发布的行情源
镜像	交换机对数据进行复制
Sailfish	连接交易所行情源的软件行情系统
SDK	接收行情数据的客户端 SDK
land_tool	配置 FPGA 的工具
land	获取 FPGA 状态和统计信息并提供监控服务
FPGA	提供行情的硬件解码服务

2.2 本地版行情架构

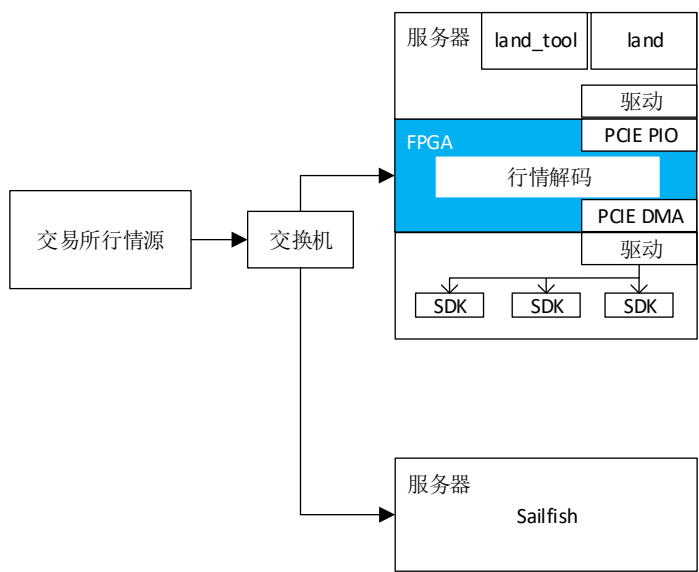


表 1.2.2 产品模块介绍

节点	说明
交易所行情源	MDGW/VDE 等行情网关发布的行情源
交换机	对数据进行组播发送，或者对数据进行复制
Sailfish	连接交易所行情源的软件行情系统
SDK	接收行情源的客户端 SDK
land_tool	配置 FPGA 的工具
land	获取 FPGA 状态和统计信息并提供监控服务
FPGA	提供行情的硬件解码服务

3 FPGA 行情版本说明

FPGA 行情包含多个版本：

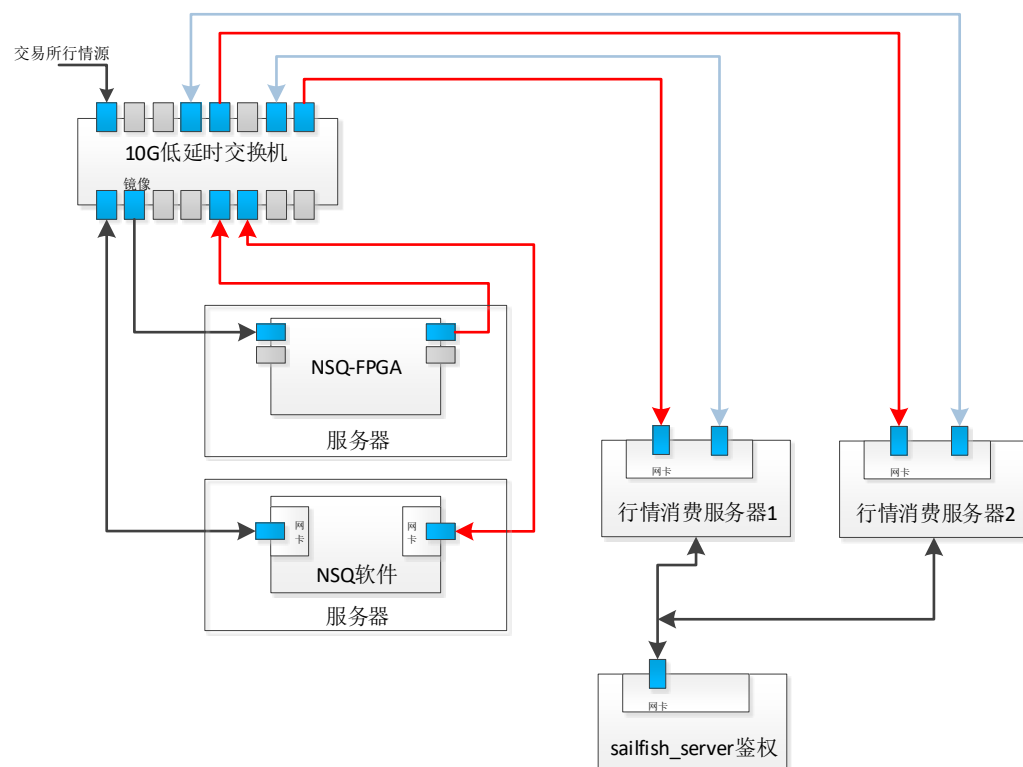
版本名称
深圳交易所 TCP 网关 L2/L1 网络版
深圳交易所组播网关 L2/L1 网络版
深圳交易所组播网关 L2/L1 本地版
上海证券交易所 TCP 网关 L2/L1 网络版
上海证券交易所 TCP 网关 L2/L1 本地版

二. 部署方案

1 典型的部署场景

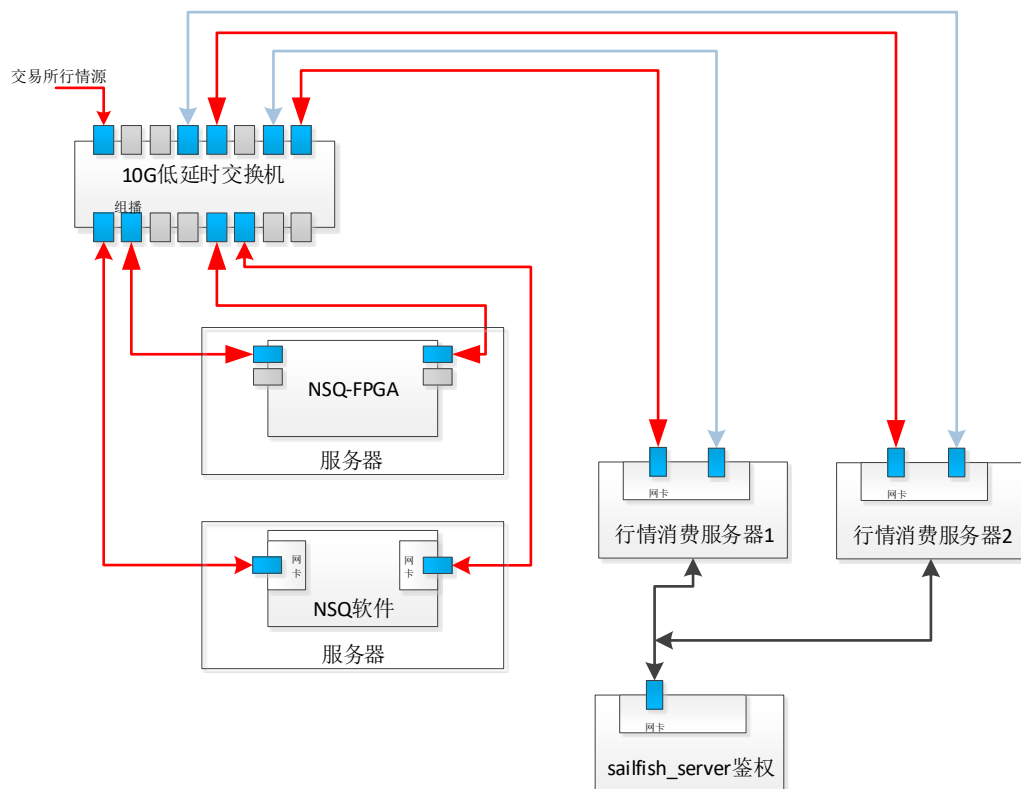
1.1 TCP 网络版

TCP 网络版的部署场景，NSQ 软件建立与行情源的 TCP 连接，该连接经过交换机镜像，下行数据复制，作为 FPGA 行情输入。TCP 下行数据复制也可以采用光分的方式。FPGA 解码后的行情数据，通过组播分发的方式，经交换机送到各客户端。



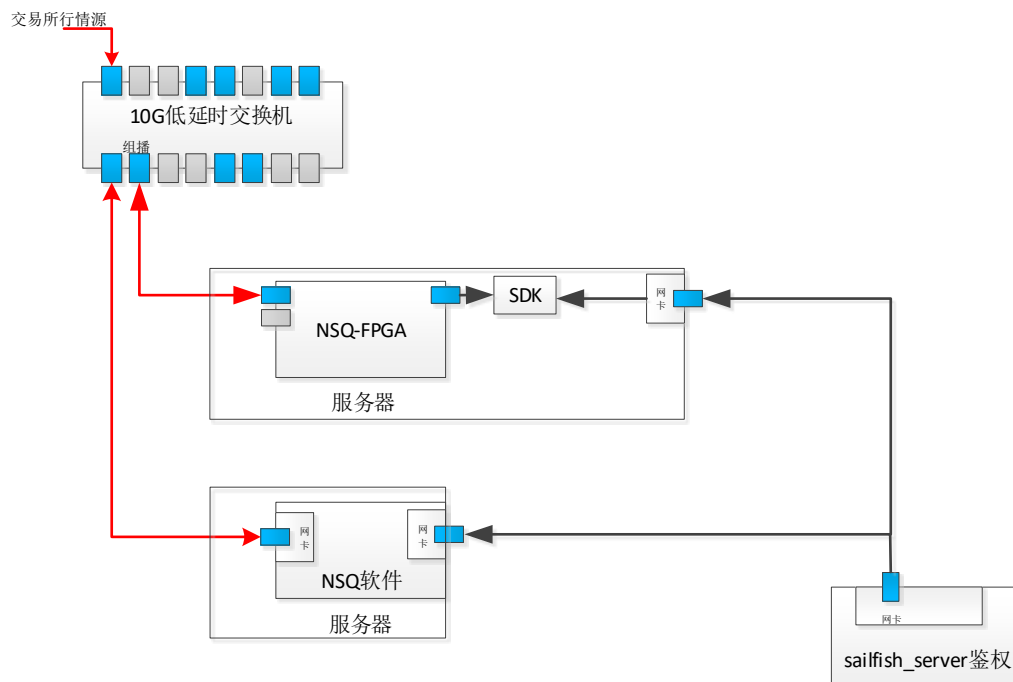
1.2 UDP 网络版

UDP 网络版的部署场景，NSQ 软件与 FPGA 同时加入行情源的组播组，接收行情源组播数据。FPGA 解码后的行情数据，通过组播分发的方式，经交换机送到各客户端。



1.3 UDP 本地版

UDP 本地版的部署场景，NSQ 软件与 FPGA 同时加入行情源的组播组，接收行情源组播数据。FPGA 解码后的行情数据通过 DMA 上送到本机的客户端 SDK。



2 部署要求

2.1 操作系统要求

对操作系统没有特殊要求，支持常用的操作系统版本，新版操作系统是否支持可与研发人员确认。

2.2 服务器硬件要求

主要考虑预留 PCIE 槽位和安装空间，其它硬件配置要求参考 NSQ。

2.3 网络环境要求

10G 局域网环境。

三. 操作说明

FPGA 极速行情系统部署主要步骤如下：

步骤 1：安装板卡（参考本章第 1 节）

步骤 2：安装驱动（参考本章第 2 节）

步骤 3：部署 land/land_tool

Tips：使用 See 发布物栈部署的方式参见本章第 6 节

运维或者测试步骤：

步骤 1：确认环境联通性，FPGA 是否能收发数据

步骤 2：开始正式接收行情前，执行 `./land_tool reset`

步骤 3：运行 `./land`。land 运行后，关闭终端界面，land 会后台运行。

当 land 后台运行时，可以提供监控服务，并且在每天零点执行 FPGA reset，清除计数器。

注意事项：

1. 如果环境设置将网卡的中断绑定在指定的 CPU，如 CPU0，那么需要把 FPGA 的中断绑定到其他 CPU。避免 FPGA 中断和网卡中断相互影响。导致网卡中断延迟或者 FPGA 状态无法正常读取。

2. 如果环境本身默认是中断负载均衡的方式，无需将 FPGA 中断绑定到指定核；

3. 如果网卡中断绑核与 FPGA XDMA 绑核冲突，在网卡流量很大的情况下，可能导致 XDMA 中断超时，无法读取 FPGA 的状态信息。

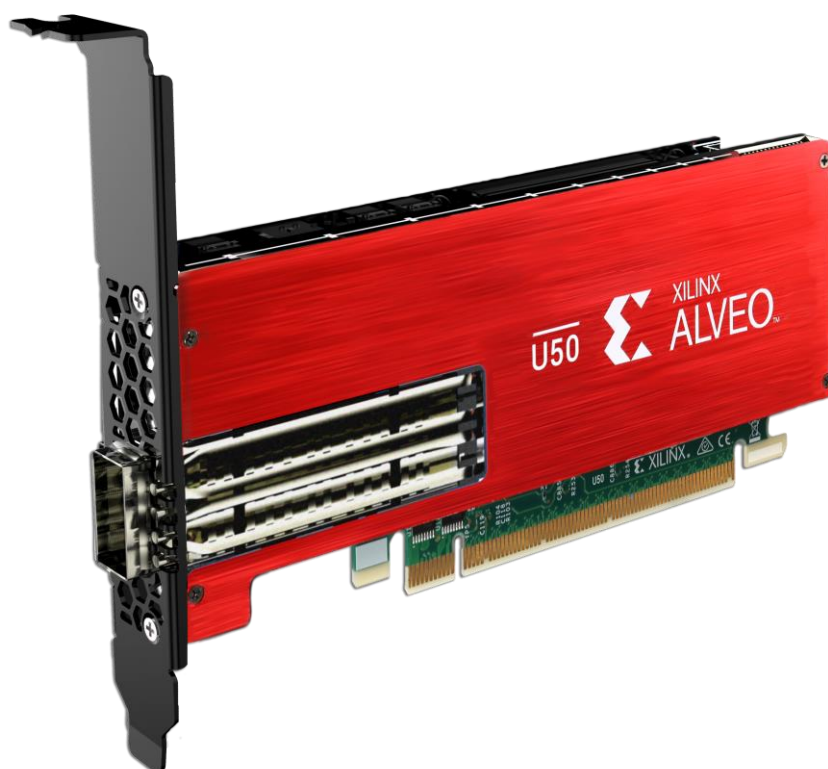
1 板卡安装

1.1 板卡介绍

恒生 FPGA 证券行情的板卡有 XILINX Alveo U200 和 XILINX Alveo U50 两种。板卡接口丰富，扩展性强，可以保证升级需求。板卡外形如下图所示：



U200 板卡展示图



U50 板卡展示图

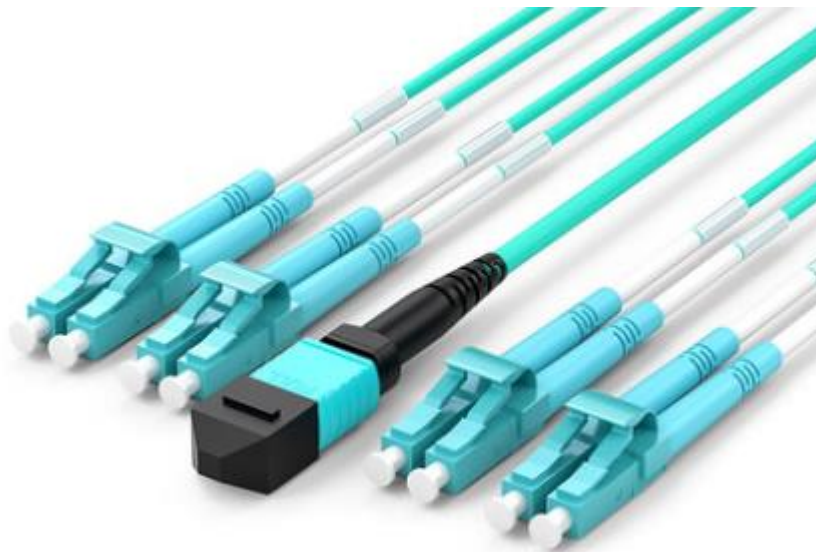
1.2 安装准备

- U200 板卡需要占用两个 PCIE 插槽宽度，U50 板卡占用一个槽位
- cisco QSFP-40G-SR4 40G 多模光模块（MPO 接口），安装在 FPGA 板卡上



针对 U200 板卡，40G 光模块插在远离 PCIE 插槽的那个光口。

- 8 芯多模光纤跳线（MPO-LC） 40G 转 4*10G 连接线



LC 头标号一般有几种情况：注意接头的颜色和线序

第一种情况，四对线分别是：1+8，2+7，3+6，4+5

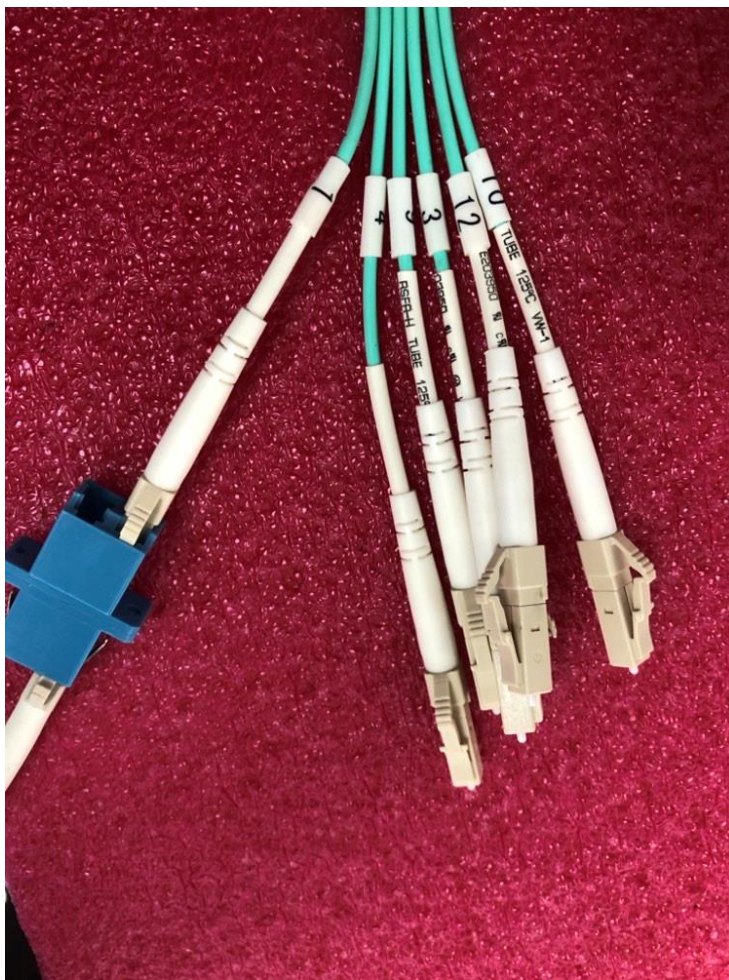


针对 U50 板卡的上证行情或者深圳行情：

1+8 为 TCP 的 RX 和 TX；

7+2 为 UDP 的 TX 和 RX；

第二种情况，四对线分别是：1+12，2+11，3+10，4+9



针对 U50 板卡的上证行情或者深圳行情：

1+12 为 TCP 的 RX 和 TX；

11+2 为 UDP 的 TX 和 RX；

以第一种情况为例(第二种情况只需把 5-8 代入 9-12):

- 针对 U200 板卡的 TCP 网关版本(2023 年之前的网络版)

通常是最小标号作为数据源输入，最大标号作为行情输出，如

1 为 TCP 的 RX，7 为 TX；

8 为 UDP 的 TX，2 为 RX；

但由于线的采购批次问题，存在一对线内 TX 与 RX 互换的情况，那么此

时最大标号变为数据源的输入，最小标号作为行情输出，如

8 为 TCP 的 RX，2 为 TX；

1 为 UDP 的 TX，7 为 RX；

- 针对 U50 板卡的上证行情或者深圳行情

通常最小标号作为数据源输入，次大标号作为行情输出，如

1+8 为 TCP 的 RX 和 TX；

7+2 为 UDP 的 TX 和 RX；

同样存在线的采购批次问题，需尝试 1 和 8 互换，同时 2 和 7 互换。

➤ 按光分器的情况下，需要准备光分器和耦合器，用于光纤与光纤的连接



按要求部署好环境后，按照要求连接好光纤。

2 驱动安装

分为 xdma 驱动和 E 驱动，需要先确认使用哪种驱动。

2.1 驱动安装和固件关系概述

2.1.1 升级固件前提

升级固件操作动作依赖已安装的驱动和正在运行的固件类型。

- 升级 xdma 固件(mcs 格式)需要已经安装好 xdma 驱动，且板卡正在运行的是 xdma 的固件。（升级命令 `./land_tool upgrade`）
- 升级 Enyx 固件(rbf 格式)需要已经安装好 E 驱动，且板卡正在运行的是 Enyx 固件。（升级方式：`enyx-firmware-manager --board=0 --file_path=/home/mcs/xilinx_u200_trade_20220412.rbf`）

2.1.2 查看当前板卡固件类型

xdma 固件：`lspci | grep Xi`

Enyx 固件：`lspci | grep Enyx`

注意，对于同一个板卡，烧写不同类型固件结果不同。比如烧写 Enyx 固件冷重启后，`lspci | grep Enyx` 会有数据，而 `lspci | grep Xi` 为空。

2.1.3 驱动名称

xdma 驱动是 `xdma.ko`

Enyx 驱动是 Enyx 字样的 rpm 包。`yum list installed | grep enyx` 如图所示

```
[root@null 0ther_u50]# yum list installed | grep enyx
enyx-cores-descriptions.noarch      2.0.1.40272-1.el7      installed
enyx-cores-lib.x86_64              6.9.0-1.el7            installed
enyx-cores-lib-devel.x86_64        6.9.0-1.el7            installed
enyx-cores-tools.x86_64            6.9.0-1.el7            installed
enyx-devframework.noarch           5.6.1-1.el7            installed
enyx-devframework-devel.noarch     5.6.1-1.el7            installed
enyx-firmware-manager.x86_64       3.4.0-1.el7            installed
enyx-hfp-core.noarch               2.12.0-1.el7           installed
enyx-hfp-lib.x86_64                2.12.0-1.el7           installed
enyx-hfp-modules.noarch            2.12.0-1.el7           installed
enyx-hfp-tools.x86_64              2.12.0-1.el7           installed
enyx-hfp-udev.x86_64               2.12.0-1.el7           installed
enyx-hw-lib.x86_64                 5.12.0-1.el7           installed
enyx-hw-lib-devel.x86_64           5.12.0-1.el7           installed
enyx-hw-tools.x86_64               5.12.0-1.el7           installed
enyx-inspector.x86_64              0.9.13r2-1.el7         installed
enyx-log-lib.x86_64                4.0.3-1.el7            installed
enyx-u200-bmc.x86_64               1.0.5r1-1.el7          installed
enyx-xupp3r-bmc.x86_64             2.2.4r1-1.el7          installed
enyx-xupvv8-bmc.x86_64            1.0.2r1-1.el7          installed
python3-enyx-hw-gen.noarch          4.0.7-1.el7            installed
[root@null 0ther_u50]#
```

2.2 xdma 驱动安装

xdma 驱动有 2 种安装方式，脚本方式安装和手动命令安装。（脚本方式是把手动命令集成）。优先脚本方式安装，如果脚本安装失败请选择手动命令安装。

2.2.1 脚本方式安装

设置环境(需要 root 权限运行)

- 执行脚本 `./install_driver.sh`，对于已经设置网卡绑核的机器，需要设置 FPGA XDMA 中断绑核参数，如 `./install_driver.sh 1`，绑核规则参见 2.3 章节
- 如果环境没有设置网卡中断绑核，无需绑核，直接执行 `./install_driver.sh` 即可。

2.2.2 手动命令安装

驱动安装需要提供 linux 系统的版本信息，比如 centos 7.4

1) 安装 xdma.ko 模块

- 拷贝驱动源文件到目标机器
- 进入 driver 文件夹，拷贝指定版本的驱动 xdma.ko
- 执行 `uname -r`

```
[root@localhost ~]# uname -r  
3.10.0-1127.el7.x86_64
```

- 拷贝 xdma.ko 至 `/lib/modules/"uname -r"结果`

如 `/lib/modules/3.10.0-1127.el7.x86_64/`

- 生成依赖
执行 `depmod`
- 执行 `sudo modprobe xdma`
- 检查驱动是否安装成功 `modinfo xdma`

```
[root@localhost bin]# modinfo xdma  
filename:      /lib/modules/3.10.0-1127.el7.x86_64/extra/xdma.ko  
license:      Dual BSD/GPL  
version:      2020.1.8  
description:   Xilinx XDMA Reference Driver  
author:       Xilinx, Inc.  
retpoline:    Y  
rhelversion:   7.8  
srcversion:    1ADA1DB6C0E1CE4C3F231F6  
alias:        pci:v00001D0Fd0000F001sv*sd*bc*sc*i*  
alias:        pci:v00001D0Fd0000F000sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00002808sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00004B28sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00004A28sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00004908sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00004828sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00004808sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006D30sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006A30sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006A28sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006930sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006928sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006830sv*sd*bc*sc*i*  
alias:        pci:v000010EEd00006828sv*sd*bc*sc*i*
```

- `ls /dev | grep xdma` 查看是否有如下 XDMA 设备

```
[root@localhost linux-kernel]# ls /dev | grep xdma
xdma0_c2h_0
xdma0_control
xdma0_events_0
xdma0_events_1
xdma0_events_10
xdma0_events_11
xdma0_events_12
xdma0_events_13
xdma0_events_14
xdma0_events_15
xdma0_events_2
xdma0_events_3
xdma0_events_4
xdma0_events_5
xdma0_events_6
xdma0_events_7
xdma0_events_8
xdma0_events_9
xdma0_h2c_0
```

如果还是没有上图显示的 XDMA 信息，重启服务器。

驱动安装成功后，需要在普通用户下运行工具。

2) 设置普通用户访问设备的权限

驱动安装成功后，需要在普通用户下运行工具，需要更改设备权限

- 进入 /etc/udev/rules.d 目录
- 新建文件名称为 40-hundsun-fpga.rules 的文件，并输入以下内容

```
KERNEL=="xdma[0-9]*", MODE="0666"
```

注意这个脚本的规则只支持最多插 10 块板卡，支持更多的板卡需要更改规则。

3) 设置 xdma 中断和网卡中断绑定到不同的 core 核

Xdma 驱动以及网卡安装完毕后，获取板卡的 pcie 的 numa 节点：

```
[root@hs-10-26-44-65 3.10.0-957.el7.x86_64]# lspci | grep Xi
d8:00.0 Serial controller: Xilinx Corporation Device 903f
[root@hs-10-26-44-65 3.10.0-957.el7.x86_64]# cat /sys/bus/pci/devices/0000\:d8\:00.0/numa_node
1
[root@hs-10-26-44-65 3.10.0-957.el7.x86_64]#
```

- Xdma 中断使用哪个核依赖条件：上述方法获取的板卡的 PCIE Numa 节点是 NUMA1，

FPGA 在 PCIE NUMA1 -> DMA 中断绑定在 NUMA0 的核

FPGA 在 PCIE NUMA0-> DMA 中断绑定在 NUMA1 的核

假设 FPGA 的 PCIE 节点在 NUMA1，则 xdma 中断使用 NUMA0，如下，另外与网卡中断使用不同的核即可。假如网卡使用 0 核，则我们可使用 NUMA0 下的除了 0 核都行，这里以使用比较空闲的 4 核为例。操作方法如下：

- 如下以 root 身份执行。
- 进入 /etc/rc.d/ 目录： `cd /etc/rc.d`
- 赋予执行权限： `chmod u+x rc.local`
- 编辑文件 rc.local 文件： `vi rc.local`
- 如下文本添加到 rc.local 末尾

```
xdmaCoreNum=10
xdmaIntNum=`(cat /proc/interrupts | grep xdma | awk 'BEGIN{FS=":"} {print $1}')`
path=`echo "/proc/irq/$xdmaIntNum/smp_affinity" | awk '{gsub(/ /,"," )}'`

echo $xdmaCoreNum > $path
```

数值 10 表示 4 核。核号与数值对应关系如下

Binary	smp_affinity (Hex)				
CPU0	0000	0000	0000	0000	1
CPU1	0000	0000	0000	0010	2
CPU2	0000	0000	0000	0100	4
CPU3	0000	0000	0000	1000	8
CPU4	0000	0000	0001	0000	10
CPU5	0000	0000	0010	0000	20

CPU6	0000 0000 0100 0000	40
CPU7	0000 0000 1000 0000	80
CPU8	0000 0001 0000 0000	100
CPU9	0000 0010 0000 0000	200
CPU10	0000 0100 0000 0000	400
CPU11	0000 1000 0000 0000	800
CPU12	0001 0000 0000 0000	1000
CPU13	0010 0000 0000 0000	2000
CPU14	0100 0000 0000 0000	4000
CPU15	1000 0000 0000 0000	8000

如果不遵照此配置，可能会导致监控和配置工具无法工作，或者当网卡流量极大时，读取缓慢或者无法正常读取。

2.3 E 驱动安装

2.3.1 安装方法

首次部署： 获取压缩包，解压后，到对应目录下执行 `yum install *rpm` 安装。

如果之前有部署，现在是升级式部署驱动，需先卸载再安装。(Tips:卸载需要在当前目录无 `enyx rpm` 包的路径下 执行 `yum remove enyx*`)。安装方式和首次部署一样。

2.3.2 部分安装错误以及解决方法

➤ 报如下错误

```

--> Package python3.x86_64 0:3.6.8-17.el7 will be installed
--> Processing Dependency: python3-libs(x86_64) = 3.6.8-17.el7 for package: python3-3.6.8-17.el7.x86_64
--> Processing Dependency: python3-setuptools for package: python3-3.6.8-17.el7.x86_64
--> Processing Dependency: python3-pip for package: python3-3.6.8-17.el7.x86_64
--> Processing Dependency: libpython3.6m.so.1.0()(64bit) for package: python3-3.6.8-17.el7.x86_64
--> Running transaction check
--> Package enyx-hfp-modules.noarch 0:2.9.4-1.el7 will be installed
--> Processing Dependency: dkms >= 2.2 for package: enyx-hfp-modules-2.9.4-1.el7.noarch
--> Package enyx-hw-lib-devel.x86_64 0:5.9.3-1.el7 will be installed
--> Processing Dependency: python3-enyx-hw-gen for package: enyx-hw-lib-devel-5.9.3-1.el7.x86_64
--> Processing Dependency: python3-jinja2 for package: enyx-hw-lib-devel-5.9.3-1.el7.x86_64
--> Package python3-libs.x86_64 0:3.6.8-17.el7 will be installed
--> Package python3-pip.noarch 0:9.0.3-8.el7 will be installed
--> Package python3-setuptools.noarch 0:39.2.0-10.el7 will be installed
--> Finished Dependency Resolution
Error: Package: enyx-hfp-modules-2.9.4-1.el7.noarch (/enyx-hfp-modules-2.9.4-1.el7.noarch)
       Requires: dkms >= 2.2
Error: Package: enyx-hw-lib-devel-5.9.3-1.el7.x86_64 (/enyx-hw-lib-devel-5.9.3-1.el7.x86_64)
       Requires: python3-enyx-hw-gen
Error: Package: enyx-hw-lib-devel-5.9.3-1.el7.x86_64 (/enyx-hw-lib-devel-5.9.3-1.el7.x86_64)
       Requires: python3-jinja2
You could try using --skip-broken to work around the problem
You could try running: rpm -Va --nofiles --nodigest

```

原因：依赖 dkms 程序包以及其他包。

➤ yum 方式安装 dkms 报如下错误

```
[root@hs-10-188-198-48 rpm]# yum install dkms
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
No package dkms available.
Error: Nothing to do
```

原因：yum 源异常。禁用即可：

```
vim /etc/yum/pluginconf.d/subscription-manager.conf enabled=0
（或者暴力删除模块 yum remove subscription-manager，不建议用）
```

➤ 之后正常安装驱动程序包。

3 监控软件

监控软件提供监控服务，包含通过定时记录日志的形式 和 see2.0 监控的方式。其中如果需要使用 see2.0 进行监控，需要在配置文件 land.ini 中做相应配置。

3.1 监控软件安装

拷贝安装文件夹到部署 FPGA 的服务器，

➤ 设置运行库目录 （libHsFPGAExporter.so libHsFpgaOS.so）

修改用户目录（~/bashrc）下 .bashrc 文件，在默认添加运行库目录，
如，

```
export LD_LIBRARY_PATH=/home/xxx/workspace/land/Lib:$LD_LIBRARY_PATH
```

其中/home/xxx/workspace/land/Lib/是运行库所在的目录

➤ 目录设置完成，关闭当前 shell 界面或者执行 source ~/.bashrc

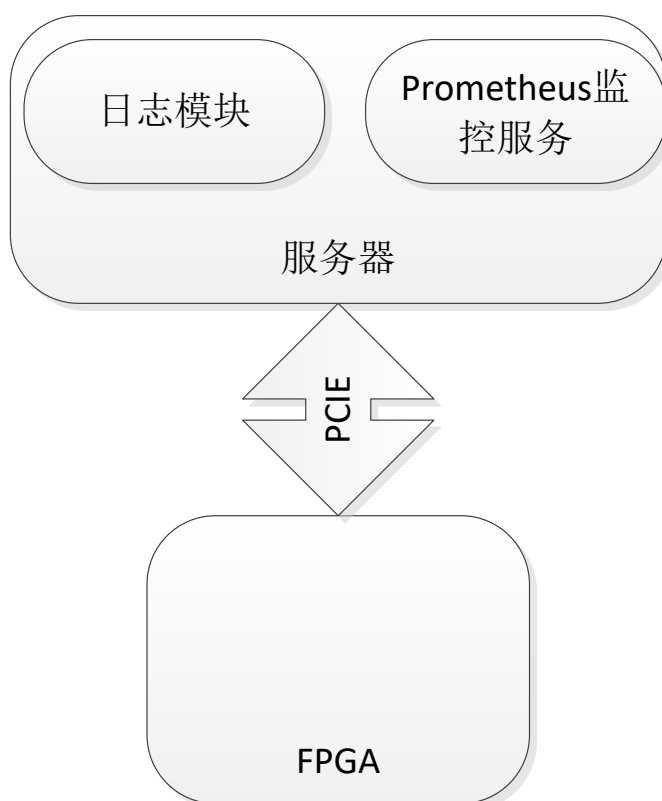
➤ 直接运行 land 程序，执行./land

此时 land 程序可以运行，但是可能配置不对导致运行后自动退出

3.2 监控软件使用

监控软件使用比较简单，配置完成后，直接运行即可，运行后直接执行 ctrl+c 会退出，如果直接关闭 shell 界面会进入后台执行。

./land 运行
监控程序框图如下：



3.2.1 监控程序的行为

监控程序运行后，根据配置文件中的信息 `record_time_len`，每固定 `record_time_len` 秒的时间通过 PCIE 读取 FPGA 的信息并写入日志文件中。

监控程序运行后，根据配置文件中配置的 `server_ip` 和 `port` 信息，提供符合 Prometheus 监视器规则的监控服务。监控项见 3.2.2 章节说明。监控项信息会发生更改。

3.2.2 监控项说明

监控分成 配置项的查看和 FPGA 状态信息监控两部分。其中配置项的查看可以远程查看 FPGA 的配置信息。

配置项的查看

配置参见配置文件说明。以下是 000 版本的示例，实际上查看监控项需要参考寄存器手册的说明。以对应版本的手册说明为准。

3.3 配置文件说明

监控软件和配置工具共用一个配置文件，配置项如下，

配置段	配置项	类型	默认值	说明
monitor				
监控配置块	port	int	8999	监控服务监听的端口号
	server_ip	string	0.0.0.0	监控服务绑定的 IP，默认 0.0.0.0 即可
	ver	string	0.0.0.1	版本信息
	url	string	空	监控 url，默认/metrics 可以不填
	gauge_cfg_name	string	configure	配置的名称
	gauge_cfg_des	string	configure	配置的描述
	gauge_stu_name	string	stu	状态的名称
	gauge_stu_des	string	stu status	状态的描述
	record_time_len	int	30	定时记录状态信息的时间，<20 时，时间间隔为 30s
device				
设备配置块	dev_write	string	/dev/xdma0_h2c_0	设备写路径，通过 ls /dev grep xdma 查看
	dev_read	string	/dev/xdma0_c2h_0	设备读路径
tcp				
Tcp 连接信息配置块	tcp_src_mac	String	00:0f:53:83:5a:d0	tcp 源端 MAC，与 MDGW 发送的网卡 MAC 一致，mac_addr_detect_switch=0 时无效
	tcp_dst_mac	String	00:0f:53:82:f9:d1	tcp 目的端 MAC，与 sailfish 接收的网卡 MAC 一致，mac_addr_detect_switch=0 时无效
	tcp_src_ip	String	8.8.20.9	tcp 源 IP，与 MDGW 发送的网卡 IP 一致
	tcp_dst_ip	String	8.8.20.13	tcp 目的 IP，与 sailfish 接收的网卡 IP 一致

	tcp_src_port	int	8098	tcp 源端口号，与 MDGW 的服务端口一致
	tcp_dst_port	string	0	tcp 目的端口号，配置成 0
udp				
UDP 信息 配置 块	udp_src_mac	string	00:0f:53:82:f0:49	udp 源 MAC 地址，通常默认即可
	udp_src_ip	string	8.10.20.90	udp 源 IP，在和 SDK 同一个 vLan 的情况下，需要同一个网段
	udp_dst_ip	string	225.0.0.177	udp 组播 IP，与 SDK 接收的组播 IP 一致
	udp_src_port	Int	8880	udp 源端口号，默认即可
	udp_dst_port	Int	13187	udp 目的端口号，与 SDK 接收的端口号一致
common				
通用 功能 适配 项	reorder_switch	Int	1	乱序重排功能开关 0：关 1：开
	enable_overflow_reset	int	0	默认不显示该项，溢出复位开关，默认值为 0，一般开了乱序一定要开这一项
	enable_snapshot_detail	int	1	委托明细功能开关 0：关 1：开
	heartbeat_switch	Int	1	心跳功能开关，默认开
	heartbeat_wait_time	int	3	心跳间隔，默认 3 秒，需要心跳功能开关打开才能生效心跳间隔
	L1_switch	int	0	L2/L1 切换开关。0：L2 1： L1
	enable_session_reset	int	1	默认不显示该项，会话切换复位开关，默认值为 1
	part_disorder_switch	Int	1	默认不显示该项，部分乱使用能开关，默认开，调试使用
	mac_addr_detect_switch	Int	0	默认不显示该项，MAC 地址检查开关，默认关
	session_sync_threshold	Int	0	同步连接需要收到的符合条件的包数量，默认 0
	session_reconnect_threshold	Int	3	切换新连接需要连续收到新连接包数量，默认 3

深圳特有配置	filter_pressb_ack_switch	Int	1	默认不显示该项，反压使能开关，默认开，调试用
上海特有配置	fcs_filter_enable	Int	0	默认不显示，接收是否存储校验
	ua3202_out_range_sw	Int	0	默认不显示，指定业务范围外的代码是否处理。打开后，不识别的代码增量不处理，只发全量。
	udp_stick_320_2_num	Int	3	组播粘包个数默认值 3，最小值 1，最大值 3，
	udp_stick_320_1_num	Int	16	组播粘包个数默认值 16，最小值 1，最大值 16
	udp_stick_311_3_num	Int	14	组播粘包个数默认值 14，最小值 1，最大值 14
	udp_stick_580_1_num	Int	20	组播粘包个数默认值 20，最小值 1，最大值 20

4 配置工具安装

4.1 配置工具安装

同 land 工具

Land_tool 用于设备升级，对于 root 用户，直接执行升级命令即可。如果有使用非 root 用户权限的需求时，需要具备特殊的权限。设置步骤如下：

4.2 配置工具使用说明

land_tool 使用参照工具的命令行提示，

```
[wufc@hs-10-26-44-65 workspace_45]$ ./land_tool
[INFO ][20210911-12:50:55][log]log is initialized successfully.
Welcome to Land tool
V1.0.0.1 Sep 11 2021 07:47:09
for help:
    ./land_tool or ./land_tool help
rest FPGA:
    ./land_tool reset
read FPGA infomation:
    ./land_tool info
config FPGA:
    ./land_tool config
read traffic status:
    ./land_tool status
or execute every n second
    ./land_tool status n
read monitor info:
    ./land_tool monitor
or execute every n second
    ./land_tool monitor n
product device info
    ./land_tool device
upgrade fpga
    ./land_tool upgrade 0000:d8:00:0 pfm_top_wrapper.mcs
fpga reload
    ./land_tool reload 0000:d8:00:0 (Attention: This may cause system reboot.)
mcast IP to target MAC addr:
    ./land_tool mip2mac 225.0.0.187
```

➤ 命令提示功能

./land_tool 或者 ./land_tool help

输出命令提示信息

➤ FPGA 重置功能

./land_tool reset

对 FPGA 进行重置，重置后 FPGA 恢复到初始状态，每次盘前需要执行一次，以保证 FPGA 处于初始化服务状态。请注意，重置的同时，会再次读取配置文件中的配置信息并写入 FPGA。

➤ FPGA 配置读取功能

./land_tool info

读取 FPGA 中的配置信息

➤ FPGA 配置功能

`./land_tool config`

实际执行效果等同与 `./land_tool reset` 重置 FPGA 的同时，重新配置 FPGA

➤ FPGA 业务状态计数器读取功能

`./land_tool status`

获取 FPGA 中的运行信息，获取行情各类型业务包数量统计信息，这里只展现最简单直观的业务包数量统计信息。

➤ FPGA 监控统计信息

`./land_tool monitor`

获取 FPGA 中，监控相关的信息，与通过监控界面获得的统计计数器相同。

➤ 查看设备信息

`./land_tool device`

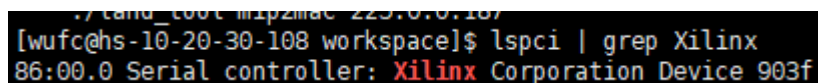
查看设备信息，版本信息等设备相关的信息。

➤ 升级功能

(1)升级 mcs 固件

升级功能首先要获取设备 ID，

1. `lspci | grep Xilinx`



```
./land_tool mip2mac 229.0.0.10/  
[wufc@hs-10-20-30-108 workspace]$ lspci | grep Xilinx  
86:00.0 Serial controller: Xilinx Corporation Device 903f
```

2.得到的 86:00.0 加上 0000: 就构成我们的 DeviceID

0000:86:00.0, 该 ID 是系统中识别 FPGA 的编号,用于升级指定的 FPGA。

./land_tool upgrade deviceID 更新包如下

./land_tool upgrade 0000:86:00.0 SZ_TCP_NET.mcs

```
20210903LLDP/ 20210907/ 20210907/ nomdclldp/
[root@hs-10-20-30-132 workspace]# ./land_tool upgrade 0000:37:00.0 APP/20210903LLDP/pfm_top_wrapper.mcs
[INFO] [20210911-12:49:39][log]log is initialized successfully.
[INFO] [20210911-12:49:39][Main]*****
[INFO] [20210911-12:49:39][Main]This command will upgrade FPGA, and you need execute reload cmd after this command.
[INFO] [20210911-12:49:39][Main]please input 'yes' to confirm or ctrl+c to exit
[INFO] [20210911-12:49:39][Main]*****
yes(no?) yes
[INFO] [20210911-12:49:44][Main]Start upgrading.....
mgmtDevice Name 0000:37:00.0
INFO: Parsing file APP/20210903LLDP/pfm_top_wrapper.mcs
INFO: **Found 498 ELA Records
Idcode byte[0] ff
Idcode byte[1] 20
Idcode byte[2] bb
Idcode byte[3] 21
Idcode byte[4] 10
Enabled bitstream guard. Bitstream will not be loaded until flashing is finished.
.....
Programming flash.....
Cleared bitstream guard. Bitstream now active.
XBFlash completed successfully. Please reboot device for flash to complete.
flash_spi delete
munmap.
[INFO] [20210911-12:54:44][Main]*****
[INFO] [20210911-12:54:44][Main]Download new image into FPGA successful.
[INFO] [20210911-12:54:44][Main]you can input yes to RELOAD image
[INFO] [20210911-12:54:44][Main]or ./land_tool reload 0000:37:00.0 later.
[INFO] [20210911-12:54:44][Main]please input 'yes' to confirm or ctrl+c to exit
[INFO] [20210911-12:54:44][Main]*****
yes(no?) ll
```

输入yes，按回车开始升级，输入其他内容或者ctrl+c退出升级

出现successfully说明写入FPGA 成功

输入yes，机器重启，重启后FPGA升级完成

如果 upgrade命令执行不想立马 reload 重启完成升级，可以输入 no 或 ctrl+c 退出。等方便重启机器的时候，输入以下命令，

./land_tool reload 0000:86:00.0 该命令执行后，FPGA 程序会更新，并且操作系统会重启。

(2) 升级 rbf 格式文件

使用系统命令

enyx-firmware-manager --board=0 --file_path=路径

比如：

enyx-firmware-manager --board=0 --

file_path=/home/mcs/xilinx_u200_trade_20220412.rbf

➤ 组播地址转 MAC 功能

./land_tool mip2mac 225.0.0.187

组播 MAC 地址遵守一定的规则，可以使用该功能计算得出。该功能只是一个小工具，不用于配置。

4.3 配置文件说明

详见监控软件配置说明

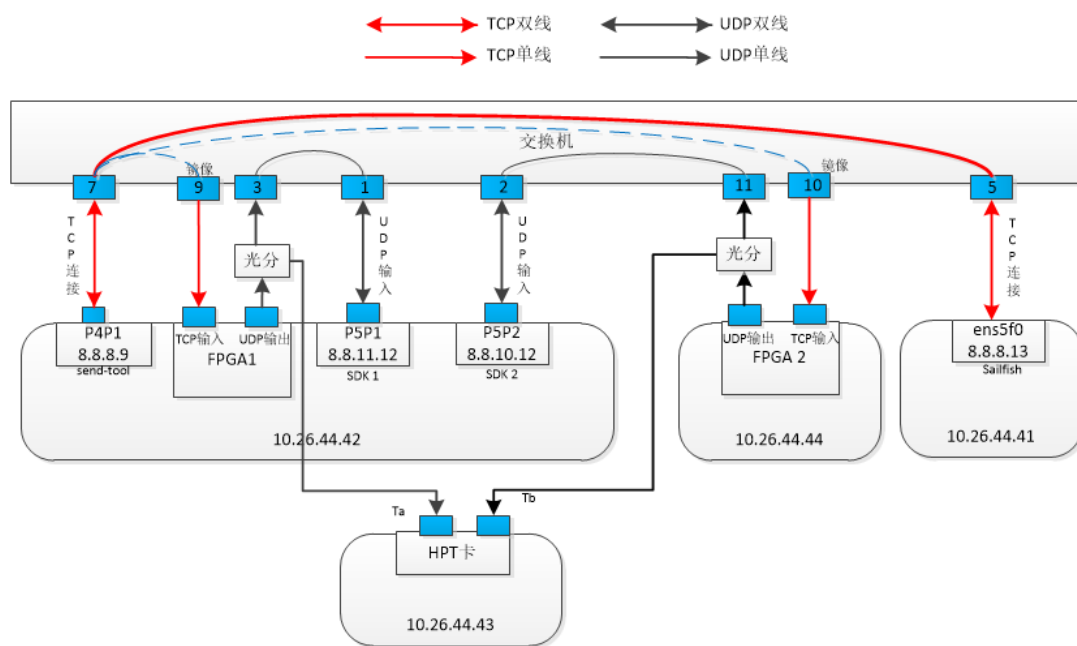
5 HPT 高精度时钟卡操作说明

POC 或者其他需要衡量 FPGA 产品性能的场景可以使用 HPT 高精度时钟卡抓包，通过 HPT 卡获取的包中包含个位数 ns 精度的时间戳。获取到的包与 tcpdump 抓到的包格式一致，可以使用 wireshark 打开查看。抓包到的 TCP 包和 UDP 包，可以使用软件团队开发的工具解析。得到统计信息。

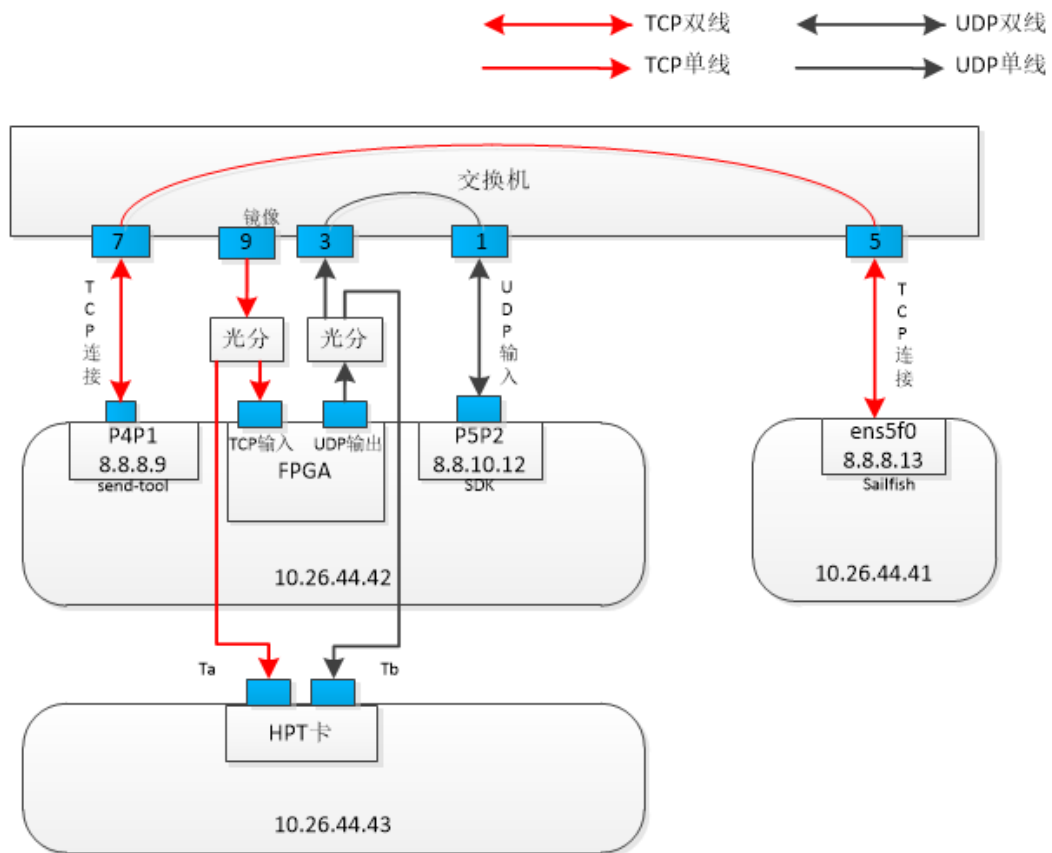
5.1 HPT 卡的两种应用场景

1. 同源对比（相同的行情源，不同的处理后，在出口处抓包计时）
2. 穿透对比（在行情源出口和解码出口分别抓包计算穿透时间）

5.1.1 同源对比



5.1.2 穿透对比



6 SEE 部署 FPGA 监控说明

SEE 部署目的是为了监控 FPGA 数据。可以更直观的查看数据且有历史查看和告警功能。



部署方法见视频：



FPGA监控see部署
方法.mp4

四. FPGA 深交所 TCP 网关 L2 行情网络版

深交所 TCP 网关 L2 行情网络版，支持深交所 MDGW 网关，接收 L2 行情并解码转发。输入 TCP 行情源，输出 UDP 协议。

1 支持业务功能范围

业务名称	是否支持（✓， ✗）	说明
逐笔成交	✓	
逐笔委托	✓	
现货快照	✓	现货（股票，基金等）集中竞价交易快照行情
指数快照	✓	
统计快照	✓	
盘后固定价	✓	

债券快照	✓	
债券成交	✓	
债券委托	✓	

五. FPGA 深交所 TCP 网关 L1 行情网络版

深交所 TCP 网关 L1 行情网络版，支持深交所 MDGW 网关，接收 L1 行情并解码转发。输入 TCP 行情源，输出 UDP 协议。

1 支持业务功能范围

业务名称	是否支持 (✓, ✕)	说明
期权	✓	

六. FPGA 上交所 TCP 网关前/网关后 L2 行情网络版/本地版

上交所 TCP 网关行情网络版/本地版，支持上交所 LDDS 网关。接收 L2 行情并转发。输入 TCP 行情源，输出 UDP 协议行情数据/DMA 数据上送。

1 支持业务功能范围

业务名称	是否支持 (✓, ✕)	说明
逐笔成交	✓	UA3201
逐笔委托	✓	UA5801
现货快照	✓	UA3202
指数快照	✓	UA3113
期权快照	✓	UA9002

债券逐笔	✓	UA3901
债券快照	✓	UA3802
现货逐笔合并	✓	UA5803
基金统计	✓	UA1502

七. FPGA 上交所 TCP 网关 L1 行情网络版

上交所 TCP 网关行情网络版，支持上交所 LDDS 网关。接收 L1 行情并转发。输入 TCP 行情源，输出 UDP 协议行情数据。

1 支持业务功能范围

业务名称	是否支持 (✓, ✕)	说明
指数快照	✓	MD001
股票快照	✓	MD002
债券分销	✓	MD003
基金快照	✓	MD004
期权快照	✓	MD301
债券快照	✓	MD201

八. FPGA 深交所 UDP 网关前/网关后 L2 行情网络版/本地版

深交所 UDP 网关 L2 行情网络版/本地版，支持深交所 MDGW 网关，接收 L2 行情并解码转发。输入 UDP 行情源，输出 UDP 协议/DMA 数据上送。

1 支持业务功能范围

业务名称	是否支持（✓， ✗）	说明
逐笔成交	✓	
逐笔委托	✓	
现货快照	✓	现货（股票，基金等）集中竞价交易 快 照行情
指数快照	✓	
统计快照	✓	
盘后固定价	✓	
债券快照	✓	
债券成交	✓	
债券委托	✓	

九. FPGA 深交所 UDP 网关 L1 行情网络版/本地版

深交所 UDP 网关 L1 行情网络版/本地版，支持深交所 MDGW 网关，接收 L1 行情并解码转发。输入 UDP 行情源，输出 UDP 协议/DMA 数据上送。

1 支持业务功能范围

业务名称	是否支持（✓， ✗）	说明
期权	✓	

十. 常见问题解决

1 SDK 收不到包的问题排查

首先，检查 FPGA 的 TCP 配置是否正确，

```

;FPGA运行参数配置
[ tcp ]
tcp_src_mac=00:0f:53:83:5a:d0 ;tcp源端MAC, 与MDGW发送的网卡MAC一致, mac_addr_detect_switch=0时无效
tcp_dst_mac=00:0f:53:82:f9:d1 ;tcp目的端MAC, 与sailfish接收的网卡MAC一致, mac_addr_detect_switch=0时无
tcp_src_ip=8.8.20.9 ;tcp源IP, 与MDGW发送的网卡IP一致
tcp_dst_ip=8.8.20.13 ;tcp目的IP, 与sailfish接收的网卡IP一致
tcp_src_port=8098 ;tcp源端口号, 与MDGW的服务端口一致
tcp_dst_port=0 ;tcp目的端口号, 配置成0

```

其次, 检查 FPGA 的组播配置是否正确,

```

[ udp ]
udp_src_mac=00:0f:53:82:f0:49 ;udp源MAC地址, 通常默认即可
udp_src_ip=8.10.20.90 ;udp源IP, 在和SDK同一个vLan的情况下, 需要同一个网段
udp_dst_ip=225.0.0.177 ;udp组播IP, 与SDK接受的组播IP一致
udp_src_port=8880 ;udp源端口号, 默认即可
udp_dst_port=13187 ;udp目的端口号, 与SDK接收的端口号一致

```

其中, UDP 的源端地址, 在和 SDK 在一个 vLan 的情况下, 需要配置在同网段。

配置检查正确, 但是依然无法收到数据时,

首先, 执行 `./land_tool info` 读取 FPGA 中的配置, 确认是否与配置文件一致,

```

***** Read Configuration *****
      Hw_Version: 4352
      FPGA_Version: 16384
      Version_Date: 2021.09.11 07:00
      Hw_Function: 2
      tcp_src_ip: 10.20.136.4
      tcp_dst_ip: 10.20.136.3
      tcp_src_port: 8100
      tcp_dst_port: 0
      tcp_src_mac: 00:0f:53:94:9c:e0
      tcp_dest_mac: 00:0f:53:94:6b:f0
      udp_src_ip: 8.8.8.10
      udp_dst_ip: 225.0.0.200
      udp_src_port: 8880
      udp_dst_port: 13200
      udp_src_mac: 00:0f:53:82:f0:49
      udp_dst_mac: 01:00:5e:00:00:c8
      lan0_reuse_en: Hex: 0x0, Dec: 0
      reset_mode: Hex: 0x0, Dec: 0
      reorder_switch: Hex: 0x0, Dec: 0
      enable_snap_order_detail: Hex: 0x0, Dec: 0
      heartbert_switch: Hex: 0x1, Dec: 1
      heartbert_wait_time: Hex: 0x12c, Dec: 300
      enable_session_reset: Hex: 0x1, Dec: 1
      enable_overflow_reset: Hex: 0x0, Dec: 0
      Ll_switch: Hex: 0x0, Dec: 0
      part_disorer_switch: Hex: 0x0, Dec: 0
      fcs_pre_filter_on: Hex: 0x0, Dec: 0
      ua3202_out_range_sw: Hex: 0x0, Dec: 0
      ua3202_out_range_sw: Hex: 0x0, Dec: 0
      udp_stick_3202_num: 3
      udp_stick_3201_num: 16
      udp_stick_3113_num: 14
      udp_stick_5801_num: 20

```

如果配置不一致，执行 `./land_tool reset`，将 `land.ini` 中的配置写入 FPGA，并将 FPGA 重置。

如果配置一致，SDK 依然收不到数据。

接下来确认 FPGA 的接收和发送端是否正常。

2 如何确认 FPGA 正在正常收包

执行 `./land_tool status`

情形一： `phy_in_link_status` 为 0，确认是线接的有问题，或者是线本身又问题。如果线序接的正确，网络版是 `phy_in_link_status` 为 3，本地版是 `phy_in_link_status` 为 1。需要特别注意。

```
phy in link status: Hex: 0x3, Dec: 3
rx_pkt_num: 0
rx_pkt_error_num: 0
tx_pkt_num: 0
tx_pkt_error_num: 0
session_lost_num_path0: 0
session_lost_num_path1: 0
drop_fcs_error_num_path0: 0
drop_fcs_error_num_path1: 0
missed_pkt_num_path0: 0
missed_pkt_num_path1: 0
step_3113_num: 0
fast_3113_num: 0
send_3113_num: 0
step_3202_num_path0: 0
step_3202_num_path1: 0
fast_3202_num_path0: 0
fast_3202_num_path1: 0
send_3202_num: 0
step_3201_num_path0: 0
step_3201_num_path1: 0
fast_3201_num_path0: 0
fast_3201_num_path1: 0
send_3201_num: 0
step_5801_num_path0: 0
step_5801_num_path1: 0
fast_5801_num_path0: 0
fast_5801_num_path1: 0
send_5801_num: 0
step_9002_num: 0
fast_9002_num: 0
send_9002_num: 0
```

情形二：

只有 `total` 计数器有值，需要再次确认 `tcp` 相关的配置是否错误

```
***** Read Monitor *****
      rx_bad_fcs: 0
      rx_packet_bad_fcs: 0
      rx_total_packets: 131
      rx_total_good_packets: 131
      tx_total_packets: 0
      session_lost_num: 0
      bram_wr_full_num: 0
      step_3202_num: 0
      step_3108_num: 0
      step_3201_num: 0
      step_3113_num: 0
      step_5801_num: 0
      recv_fast_3202_num: 0
      recv_fast_3108_num: 0
      recv_fast_3201_num: 0
      recv_fast_3113_num: 0
      recv_fast_5801_num: 0
      send_3202_num: 0
      send_3108_num: 0
      send_3201_num: 0
      send_3113_num: 0
      send_5801_num: 0
```

情形三：

正常的情形，业务计数器都有值。

```
***** Read Monitor *****
      rx_bad_fcs: 1
      rx_packet_bad_fcs: 3
      rx_total_packets: 355823
      rx_total_good_packets: 355820
      tx_total_packets: 765633
      session_lost_num: 2
      bram_wr_full_num: 0
      step_3202_num: 2704
      step_3108_num: 0
      step_3201_num: 2400
      step_3113_num: 890
      step_5801_num: 0
      recv_fast_3202_num: 364891
      recv_fast_3108_num: 0
      recv_fast_3201_num: 362428
      recv_fast_3113_num: 38314
      recv_fast_5801_num: 0
      send_3202_num: 539704
      send_3108_num: 0
      send_3201_num: 823744
      send_3113_num: 42778
      send_5801_num: 0
```

情形四：

业务计数有值，但是 tx_total_xxx 计数是 0，说明 FPGA 内部发送有问题，一般正式的版本这种情况不会出现。


```
***** Read Monitor *****
      rx_bad_fcs: 1
      rx_packet_bad_fcs: 3
      rx_total_packets: 355823
      rx_total_good_packets: 355820
      tx_total_packets: 000000 tx_total_x
                        xx 为0
      session_lost_num: 2
      bram_wr_full_num: 0
      step_3202_num: 2704
      step_3108_num: 0
      step_3201_num: 2400
      step_3113_num: 890
      step_5801_num: 0
      recv_fast_3202_num: 364891
      recv_fast_3108_num: 0
      recv_fast_3201_num: 362428
      recv_fast_3113_num: 38314
      recv_fast_5801_num: 0
      send_3202_num: 539704
      send_3108_num: 0
      send_3201_num: 823744
      send_3113_num: 42778
      send_5801_num: 0
```

3 如何确认 FPGA 正在发包

配置完成后，执行./land_tool reset 然后开始发包，执行./land_tool status 如果如下图所示，tx_total_xxx 相关的计数器值不为 0，说明 FPGA 在发包。

```
***** Read Monitor *****
      rx_bad_fcs: 1
      rx_packet_bad_fcs: 3
      rx_total_packets: 355823
      rx_total_good_packets: 355820
      tx_total_packets: 765633
      session_lost_num: 2
      bram_wr_full_num: 0
      step_3202_num: 2704
      step_3108_num: 0
      step_3201_num: 2400
      step_3113_num: 890
      step_5801_num: 0
      recv_fast_3202_num: 364891
      recv_fast_3108_num: 0
      recv_fast_3201_num: 362428
      recv_fast_3113_num: 38314
      recv_fast_5801_num: 0
      send_3202_num: 539704
      send_3108_num: 0
      send_3201_num: 823744
      send_3113_num: 42778
      send_5801_num: 0

[INFO  ][20210911-13:52:33][Srv]=====
```