



TM210 灵活 IO 卡

用户指南

发布日期 2024-09-12

前言

概述

本文档详细描述了 TM210 灵活 IO 卡的产品外观，功能以及基本配置方法。

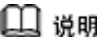
读者对象

本文档主要适用于以下人员：

- 企业管理员
- 企业终端用户

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
 须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

修改记录

文档版本	发布日期	修改说明
1.0	2024-09-12	第一次正式发布。

目 录

前言..... ii

1 产品简介.....5

1.1 概述..... 5

1.2 外观..... 5

1.3 物理规格..... 7

2 产品特性.....8

2.1 特性列表..... 8

2.2 特性介绍..... 9

3 兼容性.....12

4 配置 TM210.....13

4.1 TM210 的端口识别..... 13

4.2 驱动安装..... 15

4.2.1 获取驱动包..... 15

4.2.2 安装驱动..... 15

4.2.3 升级驱动..... 16

4.2.4 卸载驱动..... 18

4.3 TM210 固件升级..... 19

4.4 配置 VLAN..... 19

4.4.1 临时生效..... 19

4.4.2 永久生效..... 20

4.5 配置端口 bonding..... 21

4.6 使能 SR-IOV..... 22

4.7 配置网口队列数..... 23

A 缩略语.....28

1 产品简介

- 1.1 概述
- 1.2 外观
- 1.3 物理规格

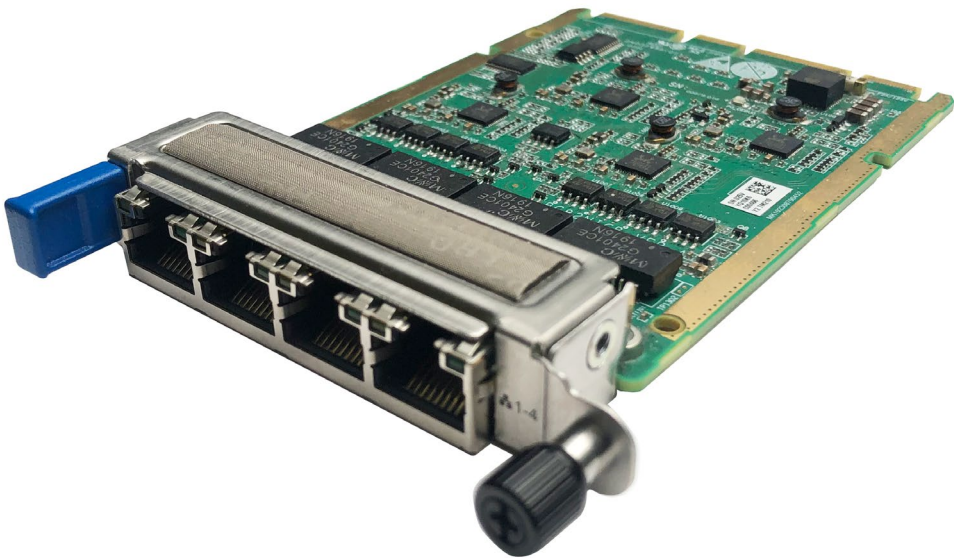
1.1 概述

TM210 网卡是一款灵活 IO 卡，提供 4 个 GE 端口来支持 NIC（Network Interface Card）应用。

1.2 外观

TM210 网卡模块的外观图 1-1 所示。

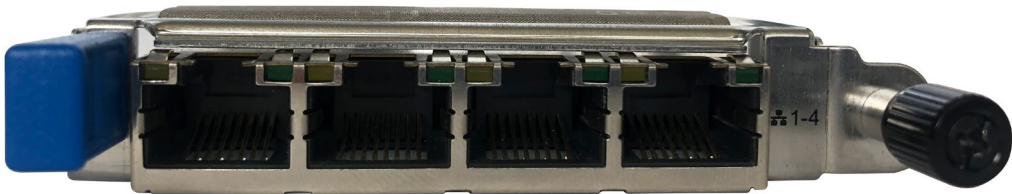
图1-1 TM210 网卡



面板

TM210 网卡面板如图 1-2 所示。

图1-2 面板



说明

网口排序从左到右依次为 1~4，其中每个网口左上角为 ACT 指示灯，右上角为 Link 指示灯。

指示灯

TM210 网卡提供 Link 和 ACT 指示灯，各个指示灯具体状态如表 1-1 所示。

表1-1 TM210 指示灯说明

指示灯类型	状态
Link 指示灯	绿色常亮：链路有 Link。

指示灯类型	状态
ACT 指示灯	绿色闪烁（2Hz）：该网口被定位。
	熄灭：链路断开。
	黄色闪烁（1Hz）：有数据传输。
	常亮：网卡无数据传输。
	熄灭：链路断开。

1.3 物理规格

TM210 的物理规格如表 1-2 所示。

表1-2 TM210 的物理规格

参数类型	参数值
接口速率	1Gbps（电口），TM210 网卡支持的工作模式及速率如表 1-3 所示。
端口数量	4 个
端口协议	Ethernet
端口形态	电口（RJ45）
芯片型号/厂家	CPU Integration/Huawei

表1-3 TM210 工作模式及速率

速率协商模式	支持的速率	不支持的速率
自协商 1000M 全双工（Full）	1000M	100M/10M
自协商 100M 全双工（Full）/半双工（Half）	100M	10M
自协商 10M 全双工（Full）/半双工（Half）	10M	-

2 产品特性

2.1 特性列表

2.2 特性介绍

2.1 特性列表

TM210 支持以下的功能特性：

- 支持 MAC VLAN 表。TM210 使用 U/M（Unicast/Multicast）VLAN 表进行 MAC+VLAN 或者 MAC 转发和过滤。
- 支持 VLAN 表。TM210 为每个 VLAN 定义了一组成员，提供 VLAN 域内过滤、广播、洪泛等功能。
- 支持流表，通过流表对报文进行过滤和转发。
- 支持流控，有效防止报文丢失。
- 支持 MTU，支持 9.5kB 的 Jumbo 帧。
- 支持 GRO（Generic receive offload），把小包封装成大包再递交给协议栈。
- 支持 VLAN 虚拟局域网。
- 支持 VxLAN，创建第 2 层逻辑网络，并将其封装在标准第 3 层 IP 数据包中。
- 支持 TSO（TCP Segment Offload）。
- 支持 RSS（receive side scaling）。
- 支持网卡队列设置。
- 支持网卡 CheckSum 功能。
- 支持网卡点灯。
- 支持网卡 SR-IOV（Single Root I/O Virtualization）。
- 支持虚拟机多队列（Virtual Machine Multi Queue）。
- 支持 DPDK（Data Plane Development Kit）。

说明

DPDK 只支持 VF，PF 因为涉及物理参数，无法支持。

默认只有 1 个队列，不支持开启 VF，需要修改 BIOS 参数支持多队列和多 function 。

- 支持网卡 PXE 功能。

2.2 特性介绍

MAC VLAN 表

TM210 使用 U/M (Unicast/Multicast) VLAN 表进行 MAC+VLAN 或者 MAC 转发和过滤。总共支持 4K+128 条表项，其中最大支持 4K+128 条单播表项，1K 组播表项。单播表项为 3K 条均分，组播表项为 1K 条全局抢占。

VLAN 表

VLAN 表为每个 VLAN 定义了一组成员，提供 VLAN 域内过滤、广播、洪泛等功能。VLAN 过滤默认是不使能的，所有的报文将不会被 VLAN 过滤掉。

流表

流表是 PPP 的一部分，主要用于对报文进行过滤和转发。基本原理是先通过用户配置或自学习的方式创建流表表项，表项形式为 key-action 形式，key 为匹配规则，包含报文的多种形式的 tuple 信息，action 为匹配之后的对报文的操作，可以为丢弃、转发到指定队列、转发到其他表进一步查表。

流控

流控通过实时了解对端设备对通信流量的要求，根据对端设备的要求来决定本设备是继续发送还是暂停发送数据包。

MTU

MTU 即 Maximum Transmission Unit (最大传输单元)，是指一种通信协议的某一层上所能通过的最大数据包大小。通常人们所说的 MTU 为 IP 层的最大传输数据大小。TM210 支持 9.5kB 的 Jumbo 帧。

GRO

GRO 是在协议栈接收报文时进行减负的一种处理方式，该方式在设计上考虑了多种协议报文。GRO 功能在接收端通过把多个相关的报文（比如 TCP 分段报文）组装成一个大的报文后再传送给协议栈进行处理，因此减少了协议栈处理报文个数，加快协议栈对报文的处理速度。

VLAN

VLAN (Virtual Local Area Network) 即虚拟局域网，是将一个物理的 LAN 在逻辑上划分成多个广播域的通信技术。TM210 最多支持 4096 个 VLAN ID，即 VLAN 0~VLAN 4095，其中 VLAN 0 和 VLAN 4095 为协议保留 VLAN ID。

VxLAN

VXLAN 是一种网络虚拟化技术，创建第 2 层逻辑网络，并将其封装在标准第 3 层 IP 数据包中。可以对二层网络在三层范围进行扩展，应用于数据中心，使虚拟机可以在互相连通的三层网络范围内迁移，而不需要改变 IP 地址和 MAC 地址。

TSO

TSO 是一种利用网卡的少量处理能力，降低 CPU 发送数据包负载的技术，需要网卡硬件及驱动的支持。

RSS

RSS 通过一定的 hash 算法实现接收方向上的负载均衡，使报文尽量平均地分配到不同队列，从而对应到不同的 CPU 核，降低因 CPU 切换调度带来的性能损失。

队列

在 NIC 领域，队列通常以 QP（Queue Pair）方式存在，单个 QP 包含 1 个 TX 队列和 1 个 RX 队列。每个 QP 映射到一块存储空间，PF/VF 通过申请 1 个或多个 QP 来实现配置和服务。当前 TM210 只支持一个队列，不可修改。

Checksum

校验功能，可以替代系统的 TCP/IP 协议栈来实现部分报文类型的校验和计算，如 IP、TCP、UDP 和 SCTP。

点灯

点灯特指对网口设备的点灯，其 LED 指示灯可用于标识不同的工作状态，例 Link 状态，电源状态，故障状态等，网口点灯功能需要软硬件配合实现。

SR-IOV

TM210 的 GE 端口支持 SR-IOV 功能。每个物理端口使能出来的 VF 可以添加给 VM（Virtual Machine）使用，与 VM 建立映射关系。

VMMQ

VMMQ 是一种 NIC 特性，它允许 VM 的流量分散到多个队列，每个队列由不同的物理处理器处理。然后，流量被传递到 VM 中的多个 LP，就像它在 vRSS 中一样，这允许向 VM 提供大量的网络带宽。

DPDK

DPDK 是一个用来进行包数据处理加速的软件库，专注于网络应用中数据包的高性能处理。

技术优点

- 通过 UIO 技术将报文拷贝到应用空间处理，规避不必要的内存拷贝和系统调用，便于快速迭代优化。
- 通过大页内存 HUGEPAGE，降低 cache miss（访存开销），利用内存多通道交错访问提高内存访问有效带宽，即提高命中率，进而提高 CPU 访问速度。
- 通过 CPU 亲和性，绑定网卡和线程到固定的 core，减少 CPU 任务切换。特定任务可以被指定只在某个核上工作，避免线程在不同核间频繁切换，保证更多的 cache 命中。
- 通过无锁队列，减少资源竞争。cache 行对齐，预取数据，多元数据批量操作。
- 通过轮询可在包处理时避免中断上下文切换的开销。

PXE

TM210 的 GE 端口支持 PXE。PXE 用于通过 Ethernet 或 IP 网络进行远程启动，远端连接 PXE Server 提供 OS 加载。

3 兼容性

TM210 支持的操作系统以及硬件的详细信息，请咨询当地销售代表。

4 配置 TM210

- 4.1 TM210 的端口识别
- 4.2 驱动安装
- 4.3 TM210 固件升级
- 4.4 配置 VLAN
- 4.5 配置端口 bonding
- 4.6 使能 SR-IOV
- 4.7 配置网口队列数

4.1 TM210 的端口识别

TM210 的端口在 OS 下识别，下面使用 CentOS 7.6 系统为例进行说明。

步骤 1 以 **root** 用户登录服务器 OS，单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 2 执行如下命令，查看 TM210 的网卡设备（PCIe function）。

lspci | grep -i eth

回显如下：

```
7d:00.0 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE RDMA
Network Controller (rev 21)
7d:00.1 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE Network
Controller (rev 21)
7d:00.2 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE RDMA
Network Controller (rev 21)
7d:00.3 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE Network
Controller (rev 21)
```

步骤 3 执行如下命令，查询到 TM210 的 4 个 NIC 端口。

ifconfig -a

TM210 的 4 个 NIC 端口以 “**enp125s0f[num]/enp189s0f[num]/eth[num]**” 的形式显示，如回显中的 “**enp125s0f0**”、“**enp125s0f1**”、“**enp125s0f2**” 和 “**enp125s0f3**”。具体端口名称与服务器的操作系统有关，操作系统不同，名称显示不同。

```
enp125s0f0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 00:01:08:00:0e:58 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp125s0f1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 128.5.251.163 netmask 255.255.255.0 broadcast 128.5.251.255
    inet6 fe80::502e:dcf:71:cad4 prefixlen 64 scopeid 0x20<link>
    ether 00:01:08:00:13:52 txqueuelen 1000 (Ethernet)
    RX packets 48529 bytes 4060749 (3.8 MiB)
    RX errors 0 dropped 6676 overruns 0 frame 0
    TX packets 185 bytes 81044 (79.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp125s0f2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 00:01:08:00:13:01 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp125s0f3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.10.10.10 netmask 255.255.255.0 broadcast 10.10.10.255
    inet6 fe80::7cd9:3aeb:beb8:4fb8 prefixlen 64 scopeid 0x20<link>
    ether 00:01:08:00:10:34 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

步骤 4 执行如下命令，查看端口信息。

ethtool -i enp125s0f[num]/enp189s0f[num]/eth[num]

如查看 **enp125s0f0** 的端口信息，命令为：**ethtool -i enp125s0f0**

回显如下：

```
driver: hns3
version: 4.19.36-vhulk1906.3.0.h356.eule
firmware-version: 1.8.12.3
expansion-rom-version:
bus-info: 0000:7d:00.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

----**结束**

4.2 驱动安装

TM210 网卡驱动默认使用操作系统自带的驱动，对于 CentOS 系统，需要手动进行驱动安装。

下面介绍 CentOS 7.6 系统下驱动安装、升级和卸载的步骤。

4.2.1 获取驱动包

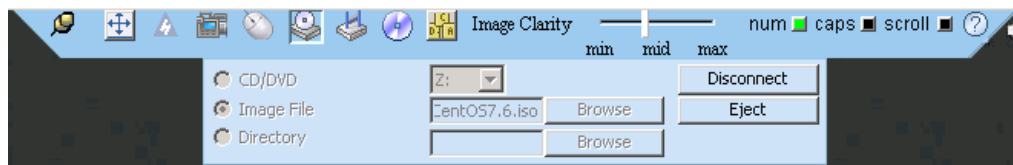
请联系技术支持获取驱动包。

4.2.2 安装驱动

步骤 1 参见 4.2.1 获取驱动包下载驱动压缩文件，CentOS 7.6 系统对应的压缩文件为“TaiShan iDriver-CentOS7.6-Driver-V100.zip”，将文件解压得到“onboard_driver_CentOS7.6.iso”镜像文件。

步骤 2 打开 iBMC 远程虚拟控制台，具体操作步骤请参考 iBMC 用户指南。

步骤 3 在 iBMC 远程虚拟控制台菜单点击光驱图标，选择“Image File”，单击“Browse”，选择已下载的驱动镜像文件“onboard_driver_CentOS7.6.iso”，单击“Connect”挂载驱动镜像文件。



步骤 4 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 5 执行如下命令，将驱动镜像文件挂载到“/mnt”路径下。

```
mount /dev/sr0 /mnt
```

说明

如服务器存在多个光驱设备，则虚拟光驱可能为 sr1、sr2 等名称，需依次进行挂载，直到确认 /mnt 目录下的文件为驱动文件。

步骤 6 执行如下命令新建路径用于保存驱动文件，例如“/root/driver”。

```
mkdir /root/driver
```

步骤 7 执行如下命令将挂载 ISO 文件后得到的驱动文件拷贝到系统下指定目录，例如“/root/driver”。

```
cp -vr /mnt/NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm /root/driver
```

步骤 8 执行如下命令编辑 dist-blacklist.conf 文件。

```
vim /usr/lib/modprobe.d/dist-blacklist.conf
```

按 Insert 键进入编辑模式，在 dist-blacklist.conf 文件末尾添加如下两行：

```
blacklist hns roce hw v2
blacklist hns_roce
```

按 ESC 键退出编辑模式，输入:wq!保存并退出。

步骤 9 执行如下命令切换到驱动文件所在目录。

```
cd /root/driver
```

步骤 10 执行如下命令安装驱动。

```
rpm -ivh --noscripts NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm
```

步骤 11 执行如下命令重新生成 initramfs 文件。

```
/sbin/depmod -a
mv /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.bak
dracut --force -v /boot/initramfs-$(uname -r).img $(uname -r)
```

回显如下类似信息，说明生成 initramfs 文件成功。

```
*** Creating initramfs image file '/boot/initramfs-4.14.0-115.el7a.0.1.aarch64.img'
done ***
```

步骤 12 输入如下命令重启系统。

```
reboot
```

步骤 13 查看驱动是否和驱动配套表中的指定版本一致。

- 安装 1.0.1 版本的驱动时，执行 **ethtool <网口名>** 命令。

显示如下字段，说明驱动和驱动配套表中的指定版本一致。

```
Current message level: 0x00000036 (54)
                        probe link ifdown ifup
```

- 安装 1.0.1 以后版本的驱动时，执行 **ethtool -i <网口名>** 命令查询驱动版本号。

例如：安装 2.0.6 版本的驱动，显示如下字段，说明驱动和驱动配套表中的指定版本一致。

```
driver: hns3
version: 2.0.6
```

----结束

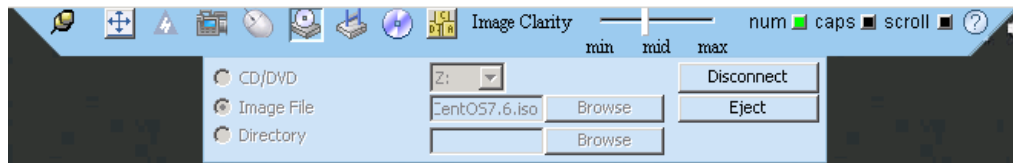
4.2.3 升级驱动

以从 1.0.1 版本驱动升级到 2.0.6 版本驱动为例说明升级驱动的操作步骤。

步骤 1 参见 4.2.1 获取驱动包下载驱动压缩文件，CentOS 7.6 系统对应的压缩文件为“TaiShan iDriver-CentOS7.6-Driver-V100.zip”，将文件解压得到“onboard_driver_CentOS7.6.iso”镜像文件。

步骤 2 打开 iBMC 远程虚拟控制台，具体操作步骤请参考 iBMC 用户指南。

- 步骤 3 在 iBMC 远程虚拟控制台菜单单击光驱图标，选择 “Image File”，单击 “Browse”，选择已下载的驱动镜像文件 “onboard_driver_CentOS7.6.iso”，单击 “Connect” 挂载驱动镜像文件。



- 步骤 4 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择 “Open Terminal” 打开命令行。

- 步骤 5 执行如下命令，将驱动镜像文件挂载到 “/mnt” 路径下。

```
mount /dev/sr0 /mnt
```

说明

如服务器存在多个光驱设备，则虚拟光驱可能为 sr1、sr2 等名称，需依次进行挂载，直到确认 /mnt 目录下的文件为驱动文件。

- 步骤 6 执行如下命令新建路径用于保存驱动文件，例如 “/root/driver”。

```
mkdir /root/driver
```

- 步骤 7 执行如下命令将挂载 ISO 文件后得到的驱动文件拷贝到系统下指定目录，例如 “/root/driver”。

```
cp -vr /mnt/NIC-hisi_eth-CentOS7.6-hns3-2.0.6-aarch64.rpm /root/driver
```

- 步骤 8 执行如下命令切换到驱动文件所在目录。

```
cd /root/driver
```

- 步骤 9 执行如下查询已安装的驱动文件名称。

```
rpm -qa | grep hisi
```

回显如下类似信息：

```
hisi_eth-1.0.1-1.centos7.6.aarch64
```

说明

如果找不到，则执行 **rpm -qa | grep hns3** 命令查询。

回显如下类似信息：

```
hns3_kernel-4.14.0-115.el7a.0.1-2.0.6-1.aarch64
```

- 步骤 10 执行如下命令卸载老版本驱动。

```
rpm -e --noscripts hisi_eth-1.0.1-1.centos7.6.aarch64
```

- 步骤 11 执行如下命令安装新版本驱动。

```
rpm -ivh --noscripts NIC-hisi_eth-CentOS7.6-hns3-2.0.6-aarch64.rpm
```

- 步骤 12 执行如下命令重新生成 initramfs 文件。

```
/sbin/depmod -a
```

```
dracut --force -v /boot/initramfs-$(uname -r).img $(uname -r)
```

回显如下类似信息，说明生成 initramfs 文件成功。

```
*** Creating initramfs image file '/boot/initramfs-4.14.0-115.el7a.0.1.aarch64.img'  
done ***
```

步骤 13 输入如下命令重启系统。

```
reboot
```

步骤 14 执行如下命令查看驱动是否升级到驱动配套表中的指定版本。

```
ethtool -i <网口名>
```

----结束

4.2.4 卸载驱动

步骤 1 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 2 执行如下查询已安装的驱动文件名称。

```
rpm -qa | grep hisi
```

回显如下类似信息：

```
hisi_eth-1.0.1-1.centos7.6.aarch64
```

说明

如果找不到，则执行 **rpm -qa | grep hns3**。

回显如下类似信息：

```
hns3_kernel-4.14.0-115.el7a.0.1-2.0.6-1.aarch64
```

步骤 3 执行如下命令卸载驱动。

```
rpm -e --noscripts hisi_eth-1.0.1-1.centos7.6.aarch64
```

步骤 4 执行如下命令重新生成 initramfs 文件。

```
/sbin/depmod -a
```

```
dracut --force -v /boot/initramfs-$(uname -r).img $(uname -r)
```

回显如下类似信息，说明生成 initramfs 文件成功。

```
*** Creating initramfs image file '/boot/initramfs-4.14.0-115.el7a.0.1.aarch64.img'  
done ***
```

步骤 5 输入如下命令重启系统。

```
reboot
```

----结束

4.3 TM210 固件升级

TM210 暂不支持固件升级。

4.4 配置 VLAN

TM210 支持 tag 和 untag 的报文通过，可在 OS 下配置 VLAN ID，或者保持 untag。

Linux 下配置 VLAN，有两种配置生效的方式：临时生效和永久生效。

- 重启网络服务影响其他业务的正常运行时，可采用 4.4.1 临时生效。
- 重启网络服务不影响其他业务的正常运行时，可采用 4.4.2 永久生效。

下面以 CentOS 7.6 系统为例介绍在 Linux 下配置 VLAN 的操作步骤。

4.4.1 临时生效

步骤 1 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 2 执行如下命令找到 NIC 端口，以 enp125s0f3 网口为例。

```
ifconfig -a
```

步骤 3 执行如下命令添加 VLAN 到端口，例如，为 enp125s0f3 添加 VLAN 100。

```
ip link add link enp125s0f3 name enp125s0f3.100 type vlan id 100
```

步骤 4 执行如下命令配置 VLAN 的 IP 地址，相当于网口子端口的 IP 地址，例如，为 enp125s0f3 端口 100 配置 IP 地址 192.168.13.200/24。

```
ip addr add 192.168.13.200/24 dev enp125s0f3.100
```

步骤 5 执行如下命令启用网卡。

```
ip link set dev enp125s0f3.100 up
```

步骤 6 执行如下命令查询配置结果。

```
ifconfig enp125s0f3.100
```

回显如下信息：

```
enp125s0f3.100: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.13.200 netmask 255.255.255.0 broadcast 0.0.0.0
    ether 00:04:45:77:0c:b6 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

说明

- 不同 VLAN 的 IP 地址要配置在不同的网段。
- 删除 VLAN 使用 **ip link delete [Device Name].[VLAN ID]** 命令，其中 *Device Name* 表示网卡设备名称（如 enp125s0f3），*VLAN ID* 表示 VLAN ID 编号（如 100）。

----结束

4.4.2 永久生效

步骤 1 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 2 执行如下命令找到 NIC 端口，以 enp125s0f3 网口为例。

ifconfig -a

步骤 3 执行如下进入网络配置文件目录。

cd /etc/sysconfig/network-scripts

步骤 4 执行如下命令复制一个“ifcfg-enp125s0f3.100”的文件。

cp ifcfg-enp125s0f3 ifcfg-enp125s0f3.100

步骤 5 执行如下命令修改“ifcfg-enp125s0f3.100”的文件中的配置。

vi ifcfg-enp125s0f3.100

按“i”修改以下备注说明的内容。

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static \\配置为静态 IP 地址
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=enp125s0f3 \\配置为 enp125s0f3.[vlan ID]
UUID=38c678e5-225d-4cc8-a7a4-dbb59ac98ef
DEVICE=enp125s0f3 \\配置为 enp125s0f3.[vlan ID]
ONBOOT=yes \\配置为 yes
IPADDR=192.168.13.100 \\配置 IP 地址
GATEWAY=192.168.13.1 \\配置网关
NETMASK=255.255.255.0 \\配置掩码
VLAN=yes \\配置为 yes
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤 6 执行如下命令重启网络服务使配置永久生效。

systemctl restart network.service

步骤 7 执行如下查询配置结果。

ifconfig enp125s0f3.100

```
enp125s0f3.100: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.13.100 netmask 255.255.255.0 broadcast 192.168.13.255
    ether 00:04:45:77:0c:b6 txqueuelen 1000 (Ethernet)
```

```
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

📖 说明

- 不同 VLAN 的 IP 地址要配置在不同的网段 (subnet)。
- 删除 VLAN 时首先需要删除配置文件 “ifcfg-enp125s0f3.100”，然后执行 **systemctl restart network.service** 命令重启网络服务。

----结束

4.5 配置端口 bonding

下面以 CentOS 7.6 为例介绍端口 bonding mode1 配置。

步骤 1 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择 “Open Terminal” 打开命令行。

步骤 2 执行如下命令找到 NIC 端口。

```
ifconfig -a
```

步骤 3 执行如下命令进入 “/etc/sysconfig/network-scripts” 目录。

```
cd /etc/sysconfig/network-scripts
```

步骤 4 创建并打开 “ifcfg-bond0”。

```
vi ifcfg-bond0
```

按 “i” 编辑文件，在 “ifcfg-bond0” 文件中添加以下内容。

```
BOOTPROTO=none
NAME=bond1
TYPE=Bond
ONBOOT=yes
IPADDR=192.168.5.122 //填写 IP 地址，作为 bonding 端口的 IP 地址
NETMASK=255.255.255.0 //填写子掩码
BONDING_MASTER=yes
DEVICE=bond0
BONDING_OPTS="mode=1 miimon=100" //填写模式及链路检测时间
```

编辑完成后按 “Esc” 退出编辑模式，输入 “:wq!” 保存并退出。

步骤 5 将 “ifcfg-enp125s0f[num]” 文件移动到 “backup” 目录进行备份。

```
mv /etc/sysconfig/network-scripts/ifcfg-eno3 /backup/ifcfg-enp125s0f2
```

```
mv /etc/sysconfig/network-scripts/ifcfg-eno4 /backup/ifcfg-enp125s0f3
```

步骤 6 创建并打开 “ifcfg-enp125s0f2” 文件。

```
vi /backup/ifcfg-enp125s0f2
```

按 “i” 编辑文件，在 “ifcfg-enp125s0f2” 中添加以下内容。

```
DEVICE=enp125s0f2
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤 7 创建并打开“ifcfg-enp125s0f3”文件。

vi /backup/ifcfg-enp125s0f3

按“i”编辑文件，在“ifcfg-enp125s0f3”中添加以下内容。

```
DEVICE=enp125s0f3
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤 8 执行如下命令加载驱动模块。

modprobe bonding

步骤 9 执行如下命令重启网络应用。

systemctl restart network.service

----结束

4.6 使能 SR-IOV

下面以 CentOS 7.6 为例介绍在 Linux 下使能 SR-IOV 的操作步骤。

步骤 1 以 **root** 用户登录服务器 OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤 2 执行如下命令找到 NIC 的端口，以 eth0 为例。

ifconfig -a

步骤 3 执行如下命令查看 NIC 端口支持的最大虚拟化网卡数量，命令执行结果会显示支持虚拟网口的最大数量。

cat /sys/class/net/enp125s0f3/device/sriov_totalvfs

步骤 4 根据实际需求创建不大于端口最大虚拟化数量的虚拟化网口，以创建 2 个虚拟化网口为例。

echo 2 > /sys/class/net/enp125s0f3/device/sriov_numvfs

步骤 5 执行如下命令查看新增 VF 对应的 PCIe 设备，其中 7d:01.0 和 7d:01.1 设备即为新增的 VF。

lspci|grep -i eth

回显如下：

```
7d:00.0 Ethernet controller: Huawei Technologies Co., Ltd. Device a222 (rev 21)
7d:00.1 Ethernet controller: Huawei Technologies Co., Ltd. Device a221 (rev 21)
7d:00.2 Ethernet controller: Huawei Technologies Co., Ltd. Device a222 (rev 21)
7d:00.3 Ethernet controller: Huawei Technologies Co., Ltd. Device a221 (rev 21)
7d:01.0 Ethernet controller: Huawei Technologies Co., Ltd. Device a22f (rev 21)
7d:01.1 Ethernet controller: Huawei Technologies Co., Ltd. Device a22f (rev 21)
```

----结束

说明

支持在 BIOS 中调整 VF 的最大个数，具体支持的 BIOS 版本和调整方法请参见《BIOS 参数参考》中“LOM 配置”章节。

4.7 配置网口队列数

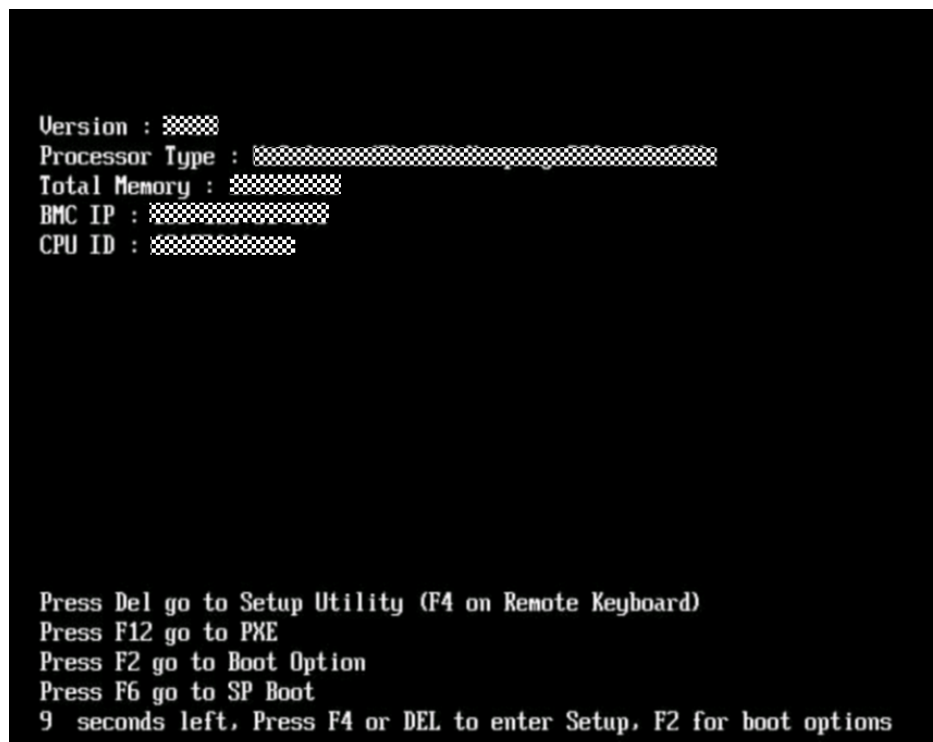
操作场景

您可通过此操作进入 BIOS 设置网口的队列个数。

操作步骤

- 步骤 1 通过服务器远程虚拟控制台登录服务器实时桌面。具体操作请参见 iBMC 用户指南。
- 步骤 2 重启服务器，当出现如图 4-1 时，按“Delete”或“F4”。

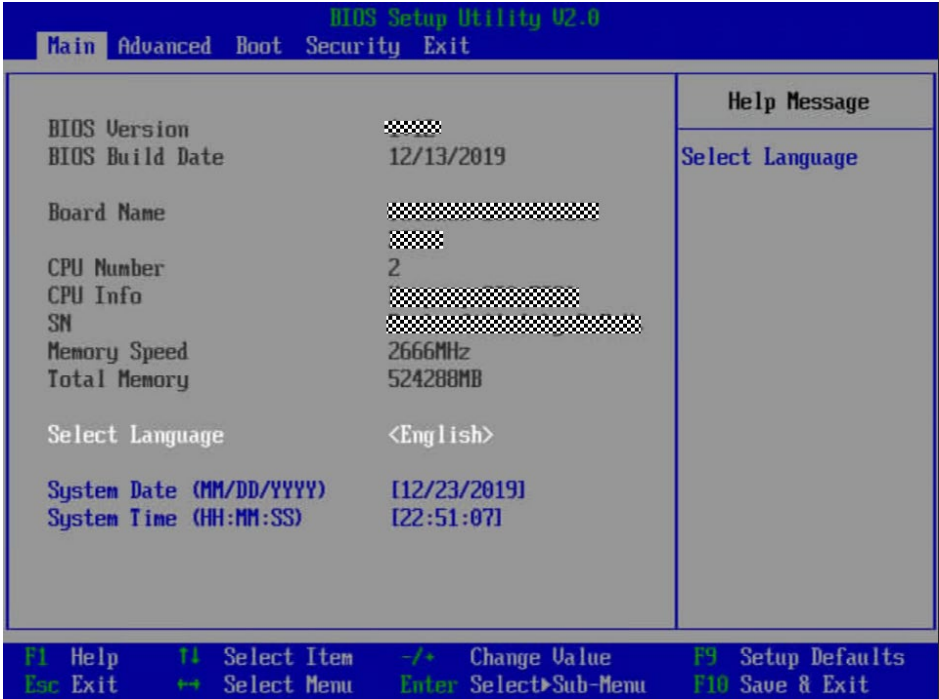
图4-1 BIOS 启动界面



步骤 3 按提示输入密码。

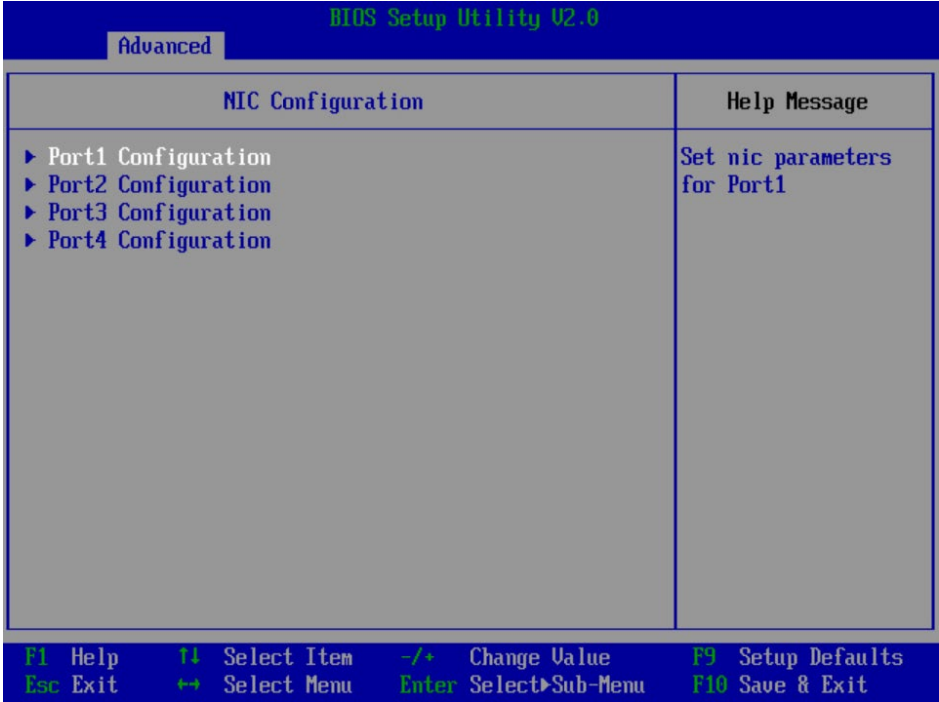
进入 BIOS 界面，如图 4-2 所示。

图4-2 BIOS 界面



步骤 4 按照 “Advanced > LOM Configuration > NIC Configuration” 顺序进入 “NIC Configuration” 界面。

图4-3 “NIC Configuration”



说明

此界面的参数根据服务器所配置的网卡动态获取，不同网卡配置显示的参数会有所不同，请以实际为准。

步骤 5 选择要配置的 Port 并按 “Enter” 。

进入 “Port Configuration” 界面，如图 4-4 所示，参数说明如表 4-1 所示。

图4-4 “Port Configuration”

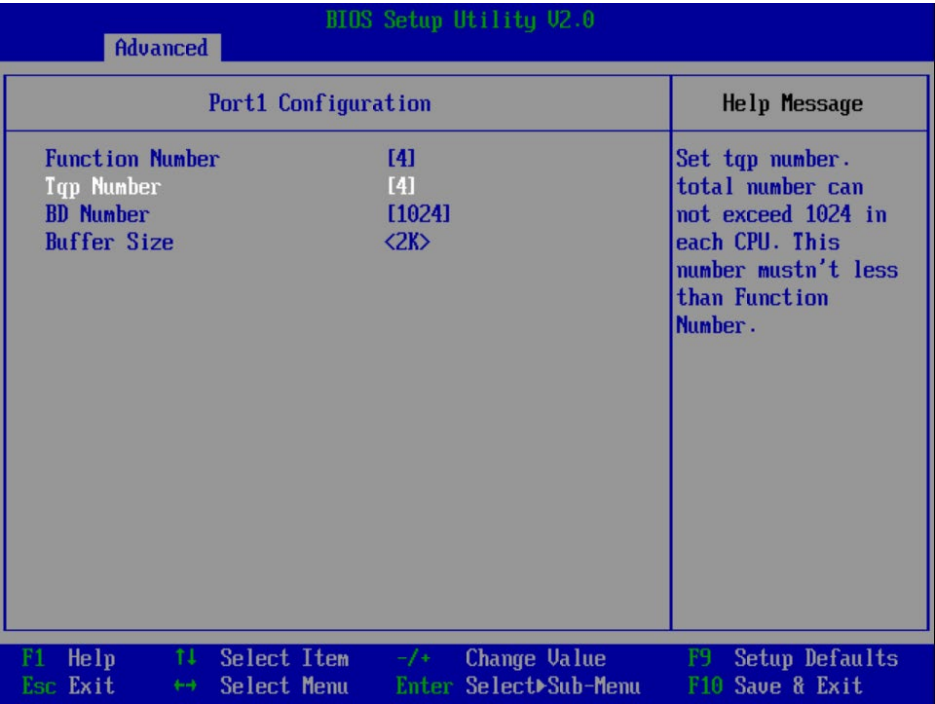


表4-1 参数说明

参数名称	功能说明	默认值
Function Number	设置网口的功能个数，每个 CPU 上的总数不超过 248，单个端口功能个数不能超过 128，且不能大于队列数。	根据板载网卡的不同，此参数的默认值会有所不同，请以实际为准。
Tqp Number	设置网口的队列数（端口所有 function 的队列总数，含 PF 和 VF），每个 CPU 上的总数不超过 1024，队列数不能小于功能个数。	根据板载网卡的不同，此参数的默认值会有所不同，请以实际为准。
BD Number	设置网口的 BD 个数，需是 8 的倍数，最小值是 72，最大值是 32760。	1024
Buffer Size	设置网口的缓冲区大小，菜单选项	2K

参数名称	功能说明	默认值
	为： • 0.5K • 1K • 2K • 4K	

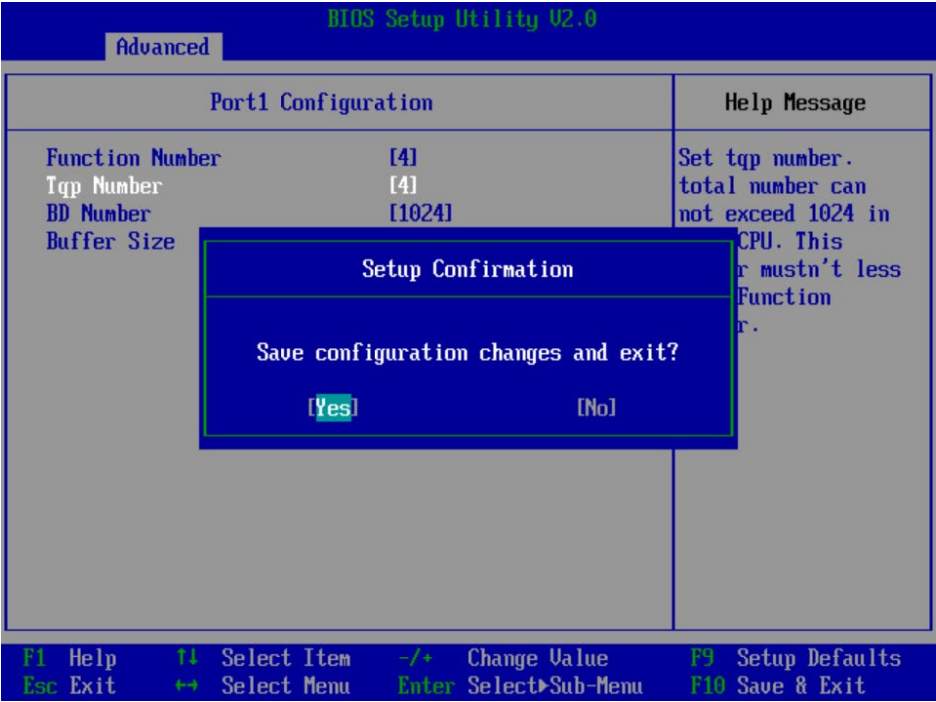
步骤 6 选择“Tqp Number”，按“Enter”输入数值。

输入完成后按“Enter”确认。

步骤 7 按“F10”弹出保存更改并退出提示，如图 4-5 所示。

选择“Yes”并按“Enter”确认退出。

图4-5 确认更改提示



----结束

A 缩略语

B		
BMC	Baseboard Management Controller	主板管理控制单元
D		
DPDK	Data Plane Development Kit	数据平面开发套件
E		
ETS	Enhanced Transmission Selection	增强型传送选择
GRO	Generic Receive Offload	通用接收卸载
I		
IP	Internet Protocol	因特网协议
M		
MAC	Media Access Control	媒体接入控制
MTU	Maximum Transmission Unit	最大传输单元
N		
NIC	Network Interface Card	网络接口卡
O		
OS	Operating System	操作系统

P		
PCIe	PCI Express	快速外围元件互连
PF	Physical Function	物理功能
PXE	Preboot Execution Environment	预启动执行环境
R		
RDMA	Remote Direct Memory Access	远端直接内存访问
RSS	Receive Side Scaling	接收端调节
S		
SR-IOV	Single Root I/O Virtualization	单主（根）输入输出虚拟化
T		
TCP	Transmission Control Protocol	传输控制协议
TSO	TCP Segmentation Offload	传输控制协议分片卸载
U		
UDP	User Datagram Protocol	用户数据报协议
V		
VF	Virtual Function	虚拟功能
VLAN	Virtual Local Access Network	虚拟局域网
VM	Virtual Machine	虚拟机
VxLAN	Virtual extensible Local Area Network	虚拟扩展局域网