

# 基于 TMS320C6678 的双网卡研究与驱动实现<sup>①</sup>

何玲玲, 王永兵

(中国电子科技集团公司 第三十二研究所, 上海 201808)

通讯作者: 何玲玲, E-mail: ruobing\_he@126.com

**摘要:** 为了将 TMS320C6678 处理器应用于与多个网络设备交互的场景中, 本文研究了该处理器的双网卡模式的工作机制, 给出了详细的驱动实现, 并在 TI 的 EVM6678L 公板上完成了测试, 测试结果证明双网卡驱动是成功的.

**关键词:** TMS320C6678; 双网卡驱动; EVM6678L

引用格式: 何玲玲, 王永兵. 基于 TMS320C6678 的双网卡研究与驱动实现. 计算机系统应用, 2018, 27(11): 236-240. <http://www.c-s-a.org.cn/1003-3254/6617.html>

## Research and Actuation Realization of Dual Network Card Based on TMS320C6678

HE Ling-Ling, WANG Yong-Bing

(The 32nd Research Institute of China Electronic Technology Group Corporation, Shanghai 201808, China)

**Abstract:** In order to apply the TMS320C6678 processor into multiple network devices interaction scenarios, this study focuses on the work mechanism of the dual network card, gives the detailed realization of its driver, and makes a test on TI's EVM6678L board. The test result proves that the double card driver is successful.

**Key words:** TMS320C6678; dual network card driver; EVM6678L

### 1 引言

随着嵌入式系统设备的广泛应用, 当设备需要与多个网络设备交互数据时, 需要设备具有多网口、多连接的功能, 同时能够解析应用规约<sup>[1]</sup>. 目前大部分对 TMS320C6678 处理器的应用场景下, 都要求网卡工作在 switch 模式, 这样两个 slave 端口始终接收到相同的数据. 所以不论是 TI 提供的 keystone 架构下的网卡参考驱动, 还是互联网上能搜索到的关于此网卡的驱动开发, 都是关于 switch 模式的. 随着 TMS320C6678 处理器的广泛应用, 越来越多的场景需要使用双网卡模式, 即 port1 和 port2 作为两个独立的网卡使用, 分别与不同的设备连接完成不同的功能, 这就需要研究基于 TMS320C6678 的双网卡驱动实现.

本文从网卡模块简介、驱动原理分析、驱动设计实现以及驱动测试等多个方面介绍了基于 TMS320C6678

的双网卡的原理和实现, 对 TI keystone 架构下的多种 DSP 处理器的双网卡模式的实现都具有参考和借鉴意义, 能极大的扩展该系列处理器的应用范围.

### 2 TMS320C6678 处理器的网卡介绍

TMS320C6678 处理器基于 KeyStone I 架构, 片上集成了一个网络协处理器 NETCP, NETCP 由千兆以太网交换子系统、包加速器 PA (Packet Accelerator) 和安全加速器 SA (Security Accelerator Engine) 组成<sup>[2]</sup>. PA 负责数据包的分发操作和修改操作, SA 负责数据包的加密和解密, 交换子系统由三端口网络交换机、MDIO 模块和 SGMII 模块组成, 其组成框图如下<sup>[3]</sup>:

系统中三端口网络交换机的 host port(即 port0) 负责交换机与网络协处理器 NETCP 之间的通信, 两个 slave port(即 port1 和 port2) 分别与两个 SGMII 模块相

① 基金项目: 国家“十三五”核高基项目 (BS170354-00)

Foundation item: National “13th Five-Year” Nuclear High Base Project of China (BS170354-00)

收稿时间: 2018-03-25; 修改时间: 2018-04-18, 2018-04-27; 采用时间: 2018-05-03; csa 在线出版时间: 2018-10-24

连,负责交换机与 SGMII 之间的数据传输. NETCP 与 主机之间的数据传输由 PKTDMA 完成.

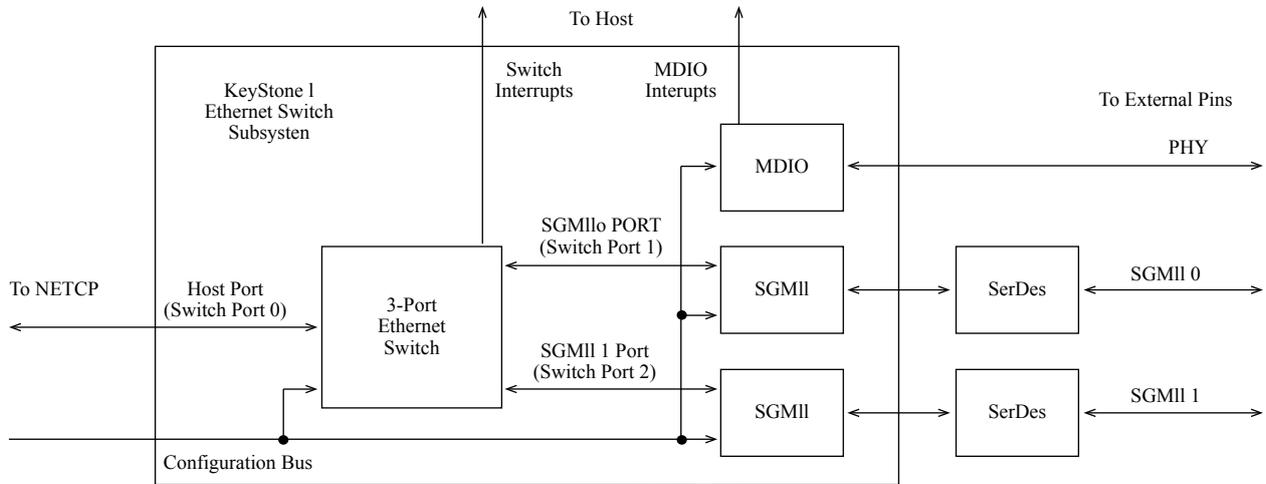


图1 以太网交换子系统框图<sup>[3]</sup>

交换机主要由以下模块组成:流媒体数据接口、MAC 控制器、MAC 接收 FIFO、信息统计模块、时间同步模块和 ALE 模块。

流媒体数据接口包括发送流媒体数据接口和接收流媒体数据接口。发送流媒体数据接口负责将数据包从 port 0 传输到 NETCP, 交换机有两个相同的发送流媒体数据接口 (TXA 和 TXB), 其中 TXA 输出从 port1 传给 port0 的数据包, TXB 输出从 port2 传给 port0 的数据包。接收流媒体数据接口负责接收来自于 NETCP 内的数据<sup>[3]</sup>。

MAC 控制器不仅负责完成交换机与 SGMII 模块之间的信号转换, 还负责与 IEEE802.3 以太网帧相关的操作; MAC 接收 FIFO 供数据传输时使用; 信息统计模块用来记录三个端口数据收发的情况, 便于网络调试; 时间同步模块的主要用途是检测时间同步事件并生成时间戳, 然后将此信息提供给主机软件进行处理。

ALE 模块处理所有接收的包, 确定每个包被发往哪个端口。当 ALE 非能时, 所有数据包都被丢弃。当 ALE 工作在 bypass 模式时, MAC 模块接收的所有数据包都只给 port0, port0 可以发送定向包给 port1 或者(和)port2。

### 3 双网卡模式下的驱动实现

#### 3.1 驱动原理

多核导航器是 Keystone 架构的核心组成部分。多核导航器使用队列管理子系统 (Queue Manager

SubSystem, QMSS) 和打包 DMA(PKTDMA) 来控制并完成高速数据包在设备内的传输。NETCP 与主机之间的数据传输由 PKTDMA 完成, PKTDMA 处理的数据是以描述符的形式存在的, 描述符存在于队列中, 由 QMSS 的 PDSP 固件完成队列中描述符的操作。当网卡硬件初始化完成后, 主机发送数据时, 从发送空闲队列分配一个描述符, 填写描述符域和负载数据后, 将描述符 PUSH 到发送队列, 队列管理器为队列提供一个层敏感 (level sensitive) 状态信号量, 负责 DMA 的调度操作, DMA 控制器最终引入相应通道的上下文, 并且开始处理包。DMA 控制器通过将数据块中的内容传输出去的方式, 来清空缓冲区。根据包大小域中指定的大小, 包中的数据全部传输完之后, DMA 会将包描述符的指针写到队列中, 这个队列在返回队列管理 (包描述符的返回队列数目域) 中被指定, 通常指定为发送空闲队列便于以后发送数据时使用。当 Rx DMA 给定的通道上开始一个包接收操作时, 这个端口会从接收空闲队列中取出一个描述符, 将数据写入描述符对应的缓存中。当整个包被接收之后, PKTDMA 将包描述符指针写入相应的接收完成队列, 用户处理完接收数据包后, 将描述符重新 PUSH 到接收空闲队列供 PKTDMA 下次接收使用。

主机发送数据时, 接收流媒体数据接口能够从接收的将要发送给以太网交换机的描述符的 PS\_FLAGS 字段中提炼出额外的控制信息。该字段的说明如表 1 所示。

从表 1 可看出, 可以在发送描述符的 PS\_FLAGS

字段中设置端口号从而将数据包定向发送给交换机的某个端口,前提是 ALE 必须工作在 bypass 模式。

表1 PS\_FLAGS 字段说明<sup>[3]</sup>

PS_FLAGS 相关的位	域	描述
3	RX_PASS_CRC	0: 数据包中不包含 CRC, CRC 需要由 MAC 模块产生 1: 数据包中已经包含 CRC, 不需要由 MAC 模块产生
2-0	TO_PORT	0: 无影响 1: 将数据包发送给 GbE 交换机的端口 1 2: 将数据包发送给 GbE 交换机的端口 2 3-7: 保留

数据接收时,不论 port1 还是 port2 接收到数据都会传输到 port0, 通过发送流媒体数据接口传输到 NETCP, 其中来自 port1 的数据通过 TXA 传输, 来自 port2 的数据通过 TXB 传输。TXA 和 TXB 不仅传输数据包, 还提供了一些额外信息, 这些信息由 PKTDMA 的接收 flow 控制。接收 flow 的配置寄存器 C 的部分字段描述如表 2 所示。

表2 接收 flow 的配置寄存器 C 的字段描述<sup>[4]</sup>

位	域	描述
31	保留位	读返回 0, 写无影响
30-28	RX_SRC_TAG_HI_SEL	接收 source tag 高字节部分选择器。指定接收包描述符的 source tag 域的位 31~位 24 的来源。 0: 不会被重写 1: 用 RX_SRC_TAG_HI 的值重写 2: 将 from_id 重写 3: 保留 4: 用 src_tag 重写 5-7: 保留
27	保留位	读返回 0, 写无影响
26-24	RX_SRC_TAG_LO_SEL	接收 source tag 低字节部分选择器。指定接收包描述符的 source tag 域的位 23~位 16 的来源。 0: 不会被重写 1: 用 RX_SRC_TAG_LO 的值重写 2: 将 from_id 重写 3: 保留 4: 用 src_tag 重写 5-7: 保留

Host packet 描述符的字段描述见表 3。

从表 2 和表 3 可以看出, 如果将网卡接收 flow 的

配置寄存器 C 的 bit26-24(RX\_SRC\_TAG\_LO\_SEL) 设置成 4, 则当端口接收到数据时, PKTDMA 控制器会将接收描述符的 word1 的 source tag-lo 字段设置成该端口的 SRC\_ID 值, 两个 slave 端口的 SRC\_ID 值是不同的。网卡接收到数据时, 根据接收描述符的 word1 的 source tag-lo 字段的值就可判断出该数据包来自哪个端口。

表3 Host packet 描述符的字段描述<sup>[4]</sup>

位	名字	描述
31-24	Source Tag -Hi	数据接收过程中, 端口的 DMA 控制器将使用 flow 配置寄存器中 RX_SRC_TAG_HI_SEL 域指定的值重写该域
23-16	Source Tag -Lo	数据接收过程中, 端口的 DMA 控制器将使用 flow 配置寄存器中 RX_SRC_TAG_LO_SEL 域指定的值重写该域
15-8	Dest Tag -Hi	数据接收过程中, 端口的 DMA 控制器将使用 flow 配置寄存器中 RX_DEST_TAG_HI_SEL 域指定的值重写该域
7-0	Dest Tag -Lo	数据接收过程中, 端口的 DMA 控制器将使用 flow 配置寄存器中 RX_DEST_TAG_LO_SEL 域指定的值重写该域

### 3.2 驱动实现

为了实现 TMS320C6678 的双网卡驱动, 需要完成硬件初始化、数据包的发送和接收等功能。尤其是要解决好下面两个问题: 数据包发送时, 交换机内部的接收流媒体数据接口需要知道它从 NETCP 接收的数据将要发给哪个端口; 接收到数据包后, 驱动需要根据接收的数据包判断其来自哪个端口。

#### (1) 网卡硬件初始化

这里需要完成跟网卡工作相关的所有硬件的初始化, 包括:

##### a) 多核导航器的初始化

多核导航器由队列管理器、PKTDMA、导航云、打包数据结构协处理器 (Packed-Data Structure Processors, PDSP) 固件等模块组成<sup>[4]</sup>。初始化时, 主要完成 PDSP 固件的下载和执行。

##### b) memory region 的初始化

完成网卡工作时使用的描述符区间的初始化工作, 包括描述符的起始地址、大小、数量、使用的内存区域的序号等的初始化。

## c) 描述符的初始化

由于数据在 NETCP 与主机之间的传输由 PKTDMA 完成, 数据包需要封装成 PKTDMA 能够识别的描述符格式. 驱动中使用了 host packet 描述符的格式.

从已经初始化的 memory region 中分配一定数量的发送空闲描述符和接收空闲描述符, 并为每个描述符分配 buffer 地址空间, 所有的发送空闲描述符被 PUSH 到发送空闲队列, 所有的接收空闲描述符被 PUSH 到接收空闲队列. 两个网卡具有不同的描述符地址和不同的发送接收空闲队列号.

## d) SGMII\_SERDES 初始化

依次配置 PLL 寄存器, 两个端口的发送配置和接收配置寄存器.

## e) SGMII 模块初始化

配置两个端口的连接模式 (master 或者 slave)、工作模式 (全双工或半双工)、连接速度、是否自动协商等.

## f) MAC 初始化

设置 MAC 地址, 虽然器件的 MAC 地址只有一个, 但是可以虚拟一个出来, 确保两个端口分别配置了不同的 MAC 地址, 除此之外, 还配置了 VLAN 和接收帧的最大长度.

## g) 三端口交换机 switch 初始化

完成端口 0 的使能启动、网卡统计功能的配置 (便于调试) 以及 ALE (Address Lookup Engine) 的配置等, 这里需要注意的是, 双网卡模式时, ALE 必须工作在 bypass 模式.

由于需要通过接收描述符中的 SRC\_ID 判断接收数据的来源, 所以分别为 port1 和 port2 设置 SRC\_ID 值. port1 和 port2 的 SRC\_ID 值即 TXA\_SRC\_ID 和 TXB\_SRC\_ID, 可通过设置寄存器 P0\_CPPI\_SRC\_ID 来设置.

## h) cpqi 初始化

完成网卡的发送通道和接收通道的使能, 接收 Flow ID、接收空闲队列和接收完成队列的配置等.

## i) PA 子系统初始化

完成 PA 固件下载、广播和组播路由配置、添加 MAC 地址到 PA PDSP 查找表等工作.

## j) 接收 flow 初始化

完成 Rx DMA 的接收完成队列、接收空闲队列、描述符类型等的配置, 需要注意配置寄存器 C 的 RX\_SRC\_TAG\_LO\_SEL 字段要设置成 4.

## (2) 数据包发送

从发送空闲队列分配一个描述符, 填写描述符域和负载数据, 注意在描述符 word2 的 Protocol Specific Flags 字段中指定数据包发给哪个端口, 然后将描述符 PUSH 到发送队列, PKTDMA 将数据包直接发送给指定的端口.

## (3) 数据包接收

当 PKTDMA 的接收完成队列接收到数据后会产生接收完成中断, 在中断服务程序中, 根据接收描述符 word1 的 source tag-lo 字段的值即可判断出数据的来源, 从而做出正确处理.

## 3.3 驱动测试

如果 TMS320C6678 开发板上的两个网络端口都通过 RJ45 接口引出来了, 那么就把它当作两个独立的普通网卡, 分别进行测试验证即可. 但是实际上有很多 DSP6678 开发板的网口只引出来了一个, 在使用中另一个端口通过特定的背板与其他以太网设备进行通信. 这里以 TIC6678 公板为例重点介绍只引出一个网络端口的情况下的双网卡测试方法.

TIC6678 公板的网卡 0 没有通过 RJ45 接口引出来, 只引出了网卡 1, 通过配套的背板可以将两块公板的网卡 0 对连起来. 当两条链路都能正常通信时, 表明双网卡能正常工作了. 具体有以下两种测试方法:

## (1) 两块公板的网卡 1 分别与两台 PC 机相连

分别为两块公板的两个网卡设置不同网段的 IP 地址, 并分别与两台 PC 机相连, IP 的具体要求以及整个测试系统的连接如图 2 所示.

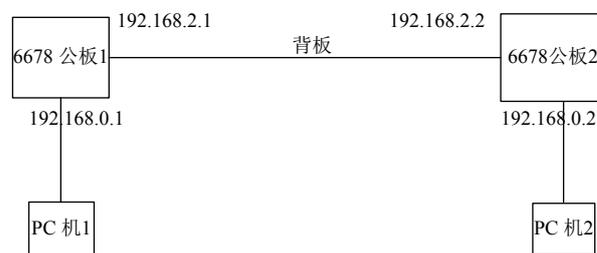


图 2 TIC6678 公板的双网卡测试系统

当同时满足以下测试结果时, 表明两块公板的双网卡均工作正常.

a) PC 机 1 可以 ping 通公板 1 的网卡 1, PC 机 1 不能 ping 通公板 2 的两个网卡;

b) PC 机 2 可以 ping 通公板 2 的网卡 1, PC 机

2 不能 ping 通公板 1 的两个网卡;

c) 公板 1 ping 不通公板 2 的网卡 1, 公板 2 ping 不通公板 1 的网卡 1;

d) 两块公板的网卡 0 互相可以 ping 通.

(2) 两块公板的网卡 1 互连

当两块公板的网卡 1 互相可以 ping 通, 网卡 0 互相可以 ping 通, 但是一块板子的网卡 1 和另一块板子的网卡 0 ping 不通时, 则双网卡工作正常.

### 3.4 测试结果

按照图 2 的 IP 要求配置了通过背板连接的两块 TIC6678 开发板的网卡, 并将它们的网卡 1 通过网线互连, 网卡 1 互相可以 ping 通, 网卡 0 互相可以 ping 通, 测试结果如图 3 和图 4 所示.

```

Serial-COM4 - SecureCRT
Serial-COM4
reworks:0>ping 192.168.2.2
reworks:0>
PING 192.168.2.2: 56 data bytes
64 bytes from 192.168.2.2: icmp_seq=0. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=1. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=2. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=3. time<1. ms
----192.168.2.2 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>ping 192.168.0.2
reworks:0>
PING 192.168.0.2: 56 data bytes
64 bytes from 192.168.0.2: icmp_seq=0. time<1. ms
64 bytes from 192.168.0.2: icmp_seq=1. time<1. ms
64 bytes from 192.168.0.2: icmp_seq=2. time<1. ms
64 bytes from 192.168.0.2: icmp_seq=3. time<1. ms
----192.168.0.2 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>

```

图 3 两块公板网卡 1 通过网线互连时, 公板 1 网卡 ping 公板 2 网卡

```

Serial-COM12 - SecureCRT
Serial-COM12
reworks:0>ping 192.168.2.1
reworks:0>
PING 192.168.2.1: 56 data bytes
64 bytes from 192.168.2.1: icmp_seq=0. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=1. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=2. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=3. time<1. ms
----192.168.2.1 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>ping 192.168.0.1
reworks:0>
PING 192.168.0.1: 56 data bytes
64 bytes from 192.168.0.1: icmp_seq=0. time<1. ms
64 bytes from 192.168.0.1: icmp_seq=1. time<1. ms
64 bytes from 192.168.0.1: icmp_seq=2. time<1. ms
64 bytes from 192.168.0.1: icmp_seq=3. time<1. ms
----192.168.0.1 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>

```

图 4 两块公板网卡 1 通过网线互连时, 公板 2 网卡 ping 公板 1 网卡

断开两块公板的网卡 1 之间的网线, 双网卡模式下网卡 1 的网络连接就断了, 测试结果如图 5 和图 6 所示.

测试结果表明本文实现的基于 TMS320C6678 的双网卡驱动工作正常. 双网卡工作模式目前已在多个项目中投入使用.

```

Serial-COM4 - SecureCRT
Serial-COM4
reworks:0>ping 192.168.2.2
reworks:0>
PING 192.168.2.2: 56 data bytes
64 bytes from 192.168.2.2: icmp_seq=0. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=1. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=2. time<1. ms
64 bytes from 192.168.2.2: icmp_seq=3. time<1. ms
----192.168.2.2 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>ping 192.168.0.2
reworks:0>
PING 192.168.0.2: 56 data bytes
Request timed out
Request timed out
Request timed out
Request timed out

```

图 5 两块公板网卡 1 之间的网线断开时, 公板 1 网卡 ping 公板 2 网卡

```

Serial-COM12 - SecureCRT
Serial-COM12
reworks:0>ping 192.168.2.1
reworks:0>
PING 192.168.2.1: 56 data bytes
64 bytes from 192.168.2.1: icmp_seq=0. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=1. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=2. time<1. ms
64 bytes from 192.168.2.1: icmp_seq=3. time<1. ms
----192.168.2.1 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0

reworks:0>ping 192.168.0.1
reworks:0>
PING 192.168.0.1: 56 data bytes
Request timed out
Request timed out
Request timed out
Request timed out

```

图 6 两块公板网卡 1 之间的网线断开时, 公板 2 网卡 ping 公板 1 网卡

## 4 结论

测试结果以及实际应用情况表明, 本文设计的基于 TMS320C6678 的双网卡驱动是成功的, 该功能的实现使得 TI KEYSTONE 1 架构下的 DSP 芯片的应用范围更加广泛而不仅仅局限于 switch 模式下的应用场景.

### 参考文献

- 徐业荣, 包明磊, 李明. 一种基于 LwIP 协议栈的双网口通信方案设计. 计算机应用与软件, 2016, 33(9): 127-129, 198. [doi: 10.3969/j.issn.1000-386x.2016.09.030]
- 邓豹. TI6678 多核 DSP 以太网接口的设计与实现. 航空计算技术, 2017, 47(4): 130-134. [doi: 10.3969/j.issn.1671-654X.2017.04.032]
- KeyStone architecture gigabit ethernet (GbE) switch subsystem user guide. SPRUGV9C. 2013.
- KeyStone architecture multicore navigator user's guide. SPRUGR9G. 2013.