

# Linux 下一种高性能数据包收发机制与实现

## Mechanism and Implementation of a High-performance Data Packet Sending and Receiving Based on Linux

刘松涛 Liu Songtao; 管鲍 Guan Bao

(武汉邮电科学研究院 武汉 430074)

(Wuhan Research Institute of Post and Telecommunications, Wuhan 430074, China)

**摘要:** 传统上 Linux 主要在内核空间收发数据包, 对于运行于用户空间的协议栈存在一次内存拷贝过程, 文章提出基于数据通道加速架构 (Datapath Acceleration Architecture, DPAA) 的用户态收发包机制, 实现内存零拷贝。介绍了 DPAA 各模块和 USDPAA 的使用方案, 对 USDPAA 内核空间和用户空间的设计进行了分析和实现。经过测试, 能够在用户态有效的收发数据包。

**Abstract:** The traditional Linux mainly sends and receives data packet in kernel space, so it has a memory copy process to protocol stack running in user space. This paper presents a mechanism of sending and receiving packet based on Datapath Acceleration Architecture, realizing the zero memory copy. Introduces each module of DPAA and the use schemes of USDPAA, analyses and realizes the design of USDPAA user space and kernel space. After testing, it is effective to sending and receiving data packet in user state.

**关键词:** 数据包; 数据通道加速架构; 内存拷贝

**Key words:** data packet; DPAA; memory copy

中图分类号: TP392

文献标识码: A

文章编号: 1006-4311(2012)15-0187-02

### 0 引言

传统的 Linux 在处理数据包时, 主要采用在内核态收发包的模式, 然后交给内核协议栈处理。对于一般的 IP 包这种处理模式没有问题, 但是对于 LTE 基站设备, 这种传统方式在处理核心网来的业务数据包, 就暴露了一些问题。因为, LTE 的协议栈一般运行在用户态环境, 那么在处理 LTE 业务包是就存在一次内存拷贝的过程, 降低了设备的吞吐量。

针对这个问题, 我们提出了一种用户态直接收发数据包的方法, 实现内存零拷贝, 加快 LTE 业务包的处理速度, 大大提高了设备的数据吞吐量。系统采用的硬件平台是基于 P4080, 操作系统为 Linux。P4080 是 Freescale 公司最新一代的多核处理器, 在其中采用了一种先进的数据通道加速架构用于数据包的收发。

### 1 DPAA 模块功能

DPAA 提供了一套硬件的数据加速平台, 主要包含 3 个大的模

块: 缓存管理 BMan、队列管理 QMan 和数据帧管理 FMan。

缓存管理 BMan 提供了内存缓存池管理功能, 可以减轻软件对内存的管理的负载。软件分配合适的缓存供不同的硬件模块使用。BMan 把这些缓存当作自由链表来维护。硬件模块从 BMan 申请缓存, 用来保存数据。当硬件模块处理完毕, 它会释放缓存还给 BMan。此外还提供了一些其他功能, 例如资源损耗跟踪。

队列管理 QMan 为网络接口硬件加速模块, 为 CPU 内核之间提供了数据队列功能。它以队列的方式来管理数据, 而且提供了各种队列相关的功能, 如拥塞管理、优先级的排队机制等。QMan 还具有包排序和恢复顺序的功能。最后, QMan 还提供了一个简化的低延迟的接口, 供软件从队列中获取数据。

数据帧管理 FMan 是一个硬件加速器, 包括包解析、流分类和 QoS 等。FMan 的首要任务是解析、分类和分发数据包 (Parse, Classify and Distribute, PCD), 并且 FMan 集合了以太网 MAC 的功能。

### 2 USDPAA 设计

USDPAA 的主要目的是实现在用户态环境下收发包, 并且实现

作者简介: 刘松涛 (1984-) 男, 湖北鄂州人, 硕士研究生, 研究方向为平台软件; 管鲍 (1977-) 男, 湖北武汉人, 博士, 研究方向为 LTE。

DAP 数据采集机的串口通信程序, 采集程序根据不同传感器的接口方式不同, 对 CC2430 单片机分别编写数据采集程序, DAP 数据采集机需要编写处理器和 CC2430 之间的串口通信程序。

3.1 无线传输模块软件设计 数据采集端软件和数据接收端软件共同组成无线传输模块的软件架构, 它包含发射和接收程序。对 CC2430 单片机射频芯片 SPI 等进行初始化设置是初始化程序的主要任务, 通过单片机的 SPI 接口, 发射程序可将打包的数据包发送至射频发生模块输出, 接收程序接收并处理终端采集的数据。图 3 即是数据采集和软件流程图。

在数据采集和传输过程中, MCU 控制器首先初始化运行和信道选择, 低功耗定时器运行准备接收信号, 等待传感器请求发送信号, 请求如合法, 则进行初始化数据采集, 并通过 CC2430 发送, 最终完成数据采集功能。

3.2 无线传感器网络采集系统软件设计 通过串口和无线传输模块, S3C6410 处理器可完成通信功能。无线传输模块将数据发送至串口, 由串口进行信息处理并将相应指令下达给无线传输模块, 通过无线方式, 无线传输模块再将指令传输给传感器模块, 由此实

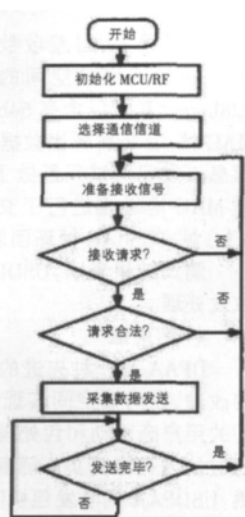


图 3 无线传输模块软件流程图

现实现无线传感器数据的监控和管理。其中, 串口初始化、发送数据函数、接收数据函数和主函数是串口应用程序的四个组成部分。

### 4 结束语

本文所设计的无线传感器网络数据采集系统基于 S3C6410 处理器、CC2430 无线通信芯片和无线传感器技术, 其性能优越, 能满足石油钻井时录井工程中的无线传感器的实时数据采集和传输。采用无线传感器网络数据采集系统在录井工程的实时数据采集应用中有很高的使用价值, 在钻井工程异常预报及石油钻井勘探中的油气发现有着重要的意义。它在未来的录井工程数据采集中必将得到广泛的应用。

本系统的优点:

- ①低功率的 ZigBee 无线传输技术降低了石油钻井现场录井数据采集时的能耗;
- ②ARM 和无线传感器网络技术使用便于现场录井数据采集的使用和管理;
- ③在录井工程现场应用无线传感器网络技术在极大程度上提高了管理效率, 同时使人工及材料成本得到了大幅度的降低。

参考文献:

- [1] 李晓维. 无线传感器网络技术. 北京: 北京理工大学出版社, 2007.
- [2] 郭渊博. ZigBee 技术与应用——CC2430 设计、开发与实践. 北京: 国防工业出版社, 2010.
- [3] 李文仲. ZigBee 无线网络技术入门与实战. 北京: 北京航空航天大学出版社, 2008.
- [4] 北京三恒星科技公司 S3C6410 中文手册. 北京: 北京三恒星科技公司.

收发包的零拷贝,以此来减小上层协议的响应时间,加快收发包的处理速度。USDPA 主要涉及到用户空间和内核空间的软件设计,其软件架构如图 1 所示。

2.1 USDPA 的使用方案 USDPA 包括四种典型的应用实例:①QMan 将 FMan 与内核协议栈直连;②QMan 将 FMan 与 USDPA 直连;③QMan 将 FMan 与内核协议栈和 USDPA 均相连。在入口侧,FMan 选择数据帧的目的,将其入队相应的 QMan queue,不同的 queue 分别与内核协议栈和 USDPA 相连,这需要 PCD 支持;④USDPA 直接与内核协议栈相连,不需要 FMan 参与。如图 2 所示。

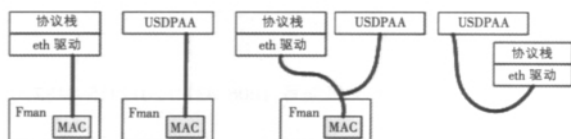


图 2 USDPA 应用实例

2.2 内核空间设计 内核部分的设计主要为:①申请一大块物理地址连续的内存用于 USDPA 收发包,并且通过 mmap 映射到用户空间,实现零拷贝;②使用 UIO 映射 DPAA 寄存器到用户空间供上层直接配置;③device tree 提供对 USDPA 相应字段的解析。

2.2.1 内存映射 FMan 和 QMan 通过 DMA 的方式使用分配给 BMan 的内存,这是属于硬件上的操作,需要这些内存物理地址上连续。因此在系统启动阶段,采用 LMB 的方式从 DDR 上保留一段内存,内存的地址范围位于内核内存的高端地址上。这段内存分配成功后,需要将其地址空间记录到 TLB1 中,P4080 的 TLB1 支持多种大小的 buffer,这样可以保证应用需要访问这段空间时,后续的操作不会发生 page-missing。这里还提供了 ioctl 接口给应用获取这段内存的起始地址和大小。

2.2.2 DPAA 寄存器的 UIO 为了避免用户空间收发包时频繁调用 ioctl 来访问 DPAA 寄存器,这里采用了 UIO 接口。

QMan 和 BMan 通过 UIO 将 cache-enable、cache-inhibited 寄存器暴露给用户空间的应用。驱动程序在用户态可以直接操作这些映射后的寄存器,来达到配置和收发包的目的。

2.2.3 设备树 在 dts 文件上,添加“fsl, usdpaa-portal”属性字段,以此来表示 QMan portal(一种软件意义上的逻辑接口,用于访问 QMan 设备)和 BMan Portal 是否支持 USDPA portal。

添加了这个属性的 portal,在内核进行设备初始化时不会对其初始化,而是在用户态驱动加载时才对其初始化。换句话说,如果某个 portal 用于在用户态收发包,它就不允许在内核态收发包。

2.3 用户空间设计 USDPA 的用户空间部分需要负责收发包,管理 BMan 和 QMan,并且需要提供对上层的接口。USDPA 的用户空间实现可以分为三层:us-driver、PPAC (package-processing application core)和 PPAM(package-processing application module)。

2.3.1 us-driver 该部分主要操作 DPAA 寄存器,负责配置 BMan 和 QMan,实现框架与内核驱动大致一样。该部分还封装了内核接口,比如 percpu 变量、spinlock 和 local\_irq\_enable 等。

us-driver 主要集成了 DMA 内存管理,因为内核空间用于收发包的内存被映射到了用户空间,内核没有管理这段内存,所以 us-driver 需要自己管理这段内存,而不能直接使用 malloc、free 等系统调用。us-driver 实现了四个 DMA 接口管理这段内存: dma\_mem\_malign、dma\_mem\_free、dma\_mem\_ptov、dma\_mem\_vtop。dma\_mem\_setup 将这段内存映射到用户空间。针对这段内存,驱动划分了三种不同大小的内存池来接收不同长度范围的数据,分别为 700、1100 和 2200bytes,这三种内存池数量分别为 0x4000、0x1000 和 0x1000。

2.3.2 PPAC PPAC 实现 USDPA 的基本功能,包括设备初始化、流控处理、CLI 实现、线程和内存管理等。PPAC 封装了许多底层驱动的细节,因此 USDPA 的具体应用都可以基于 PPAC,通过注册相应的回调函数能使应用更容易实现。还可以根据具体的应用优化 PPAC,提高 USDPA 的性能。PPAC 的处理流程如图 3 所示。

这里比较关键的策略是 select(irq 模式)和 polling 之间的切换机制,这里是这样实现的:首先进入 polling 模式,如果一段时间内没有收到包(比如连续 2000 次循环),那么切换到 irq 模式。一旦在 irq 模式下有中断到达(不一定有数据到,绝大多数情况是收包就中断),则马上切换到 polling 模式。这跟内核 NAPI 方式有所区别,NAPI 在连续收到一定数量的包后会切换到 irq 模式。

2.3.3 PPAM PPAC 实现 USDPA 的基本功能,而 PPAM 则主要实现上层应用的具体逻辑业务,即实现收发包的接口。它提供的接口有 ppam\_rx\_hash\_cb()、ppam\_tx\_confirm\_cb()、ppac\_send\_frame()、ppac\_drop\_frame()。

### 3 实验测试

这里的设计采用的第二种使用方案,将修改后的 USDPA 内核代码和用户空间收发数据包代码编译成模块和可执行程序,下载到目标机后加载驱动模块和运行可执行程序,通过交换机接入到网络。此时运行 USDPA 的网口上没有协议栈,因此在 PC 上使用发包工具发送数据给目标机,进行收发包测试。测试输出结果如图 4 所示。

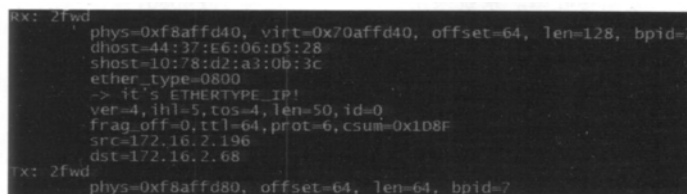


图 4 USDPA 收发包测试结果

从 7 号 portal 接收数据,存放在物理地址为 0xf8affd40 的内存中,映射到用户空间的虚拟地址为 0x70affd40。数据总长度为 128bytes,其中包含有 64bytes 的处理信息,因此偏移了 64bytes 指向 MAC 帧。首先解析出数据包的以太网帧头信息,然后解析出 IP 包头信息。发包测试只是做了环回,把收到的包的 IP 地址进行交换,同时 MAC 地址也进行了交换,然后偏移 64bytes 指向 0xf8affd80 的 MAC 帧,把 MAC 帧环回回去。

测试结果表明,USDPA 能够在用户空间有效的进行数据包收发处理。

### 4 结束语

DPAA 是一种先进的数据包处理技术,通过对 DPAA 软件架构的改进,在内核空间实现内存映射和寄存器的 UIO 设计,在用户空间的驱动和包处理设计,可以实现 USDPA 进行用户态收发包时的内存零拷贝。随着网络的全 IP 化,网络业务都将在 IP 上承载,USDPA 的收发包机制将能够有效的提高设备的数据吞吐率。

#### 参考文献:

- [1]路冉冉,郝静如,李天剑.嵌入式系统的以太网接口设计及 linux 驱动[J].电子技术,2009,46(5):35-37.
- [2]Jonathan Corbet, Alessandro Rubini, Greg Kroah-hartman. Linux 设备驱动程序[M].第 3 版.北京:中国电力出版社,2006.
- [3]王齐.Linux PowerPC 详解——核心篇[M].第 1 版.北京:机械工业出版社,2007.9.
- [4]董志国,李式巨.嵌入式 linux 设备驱动程序开发[M].计算机工程与设计,2006.10,27(20):3737-3740.