

# 在 TMS320DM642 上实现嵌入式 TCP/IP 协议栈

Embedded Solution of TCP/IP Protocol Stack Based on TMS320DM642

王 平\* 孙乐美\*\*

WANG Ping SUN Le - mei

**摘 要** 实现了一个运行在数字信号处理器 TMS320DM642 上的嵌入式 TCP/IP 协议栈。对 TCP/IP 协议中的 IP 协议、ARP 协议、UDP 协议进行分析,完成基于 TCP/IP 协议的嵌入式网络系统。

**关键词** DSP TMS320DM642 TCP/IP 协议栈 嵌入式系统

**Abstract** One embedded TCP/IP protocol stack running on digital signal processor TMS320DM642 is realized. IP protocol, ARP protocol, UDP protocol of TCP/IP protocol stack are analyzed. One embedded network system based on TCP/IP protocol stack is accomplished.

**Keywords** : DSP TMS320DM642 TCP/IP Protocol Stack Embedded System

## 引言

随着嵌入式系统应用范围的不断扩展及网络应用的日益普及,使得越来越多的嵌入式系统需要支持网络功能。TCP/IP 是目前一种被广泛采用的网络协议。TCP/IP 是一套把因特网上的各种系统互连起来的协议组,已成为事实上最常用的网络标准之一。本文对 TCP/IP 协议中的 IP 协议、ARP 协议和 UDP 协议进行了分析,利用 DSP 的专用汇编语言实现了自己的通信协议。由此可使 DSP 直接对 Ethernet 发来的数据包解分析,并能按照 TCP/IP 协议正确打包和发送,最终实现 DSP 与 PC 机的数据通信,成功地完成了视频数据的采集和发送,实现了 DSP 与 PC 机的视频数据传输。

## 1 TCP/IP 协议分析

一般在嵌入式系统中实现的 TCP/IP 协议都是面向数据采集和传输的,所以大部分实现都是 IP 协议、ARP 协议、UDP 协议或者是 TCP 协议。本文完成的是 IP 协议、ARP 协议和 UDP 协议。一般认为 TCP/IP 为四层协议,实现的协议栈结构如图 1 所示。

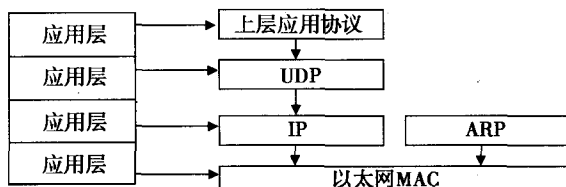


图 1

DSP 程序自下而上实现以下协议:

①媒介访问控制 MAC(Media Access Control),向以太网收发数据。传送的数据格式为 Ethernet 数据帧格式。Ethernet 帧的长度是可变的,但都大于 64 字节,小于 1518 字节。它包括头部、数据和尾部三部分。8 字节的前导用于帧同步, CRC 域用于帧校验。目的地址和源端地址是指网卡的物理地址(MAC 地址),具有唯一性。

②地址解析协议 ARP(Address Resolution Protocol),使得物理地址和 IP 地址可以对应起来。在 Ethernet 上,使用地址解析协议 ARP 来实现 IP 地址到 MAC 地址的动态转换。ARP Request(ARP 请求)和 ARP Response(ARP 响应)类型都是 0X0806。ARP 报文格式包括硬件类型、协议类型、硬件地址长度、协议地址长度、操作字段、源端物理地址、源端 IP 地址、目的物理地址和目的 IP 地址。

③Internet 网核心协议 IP(Internet Protocol),实现 IP 网络上的数据传输的基础。IP 协议提供了一种高效、不可靠和无连接的传输方式。IP 报文是 IP 协议的基本处理单元,是由报头和数据两部分组成的。IP 报头包括 20 字节的固定部分和变长的选项部分。20 字节的固定部分包括 IP 协议的版本号、IP 首部长、服务类型、报文总长度、标识符、片偏移、报文生存时间、首部校验和以及源端 IP 地址和目的端 IP 地址。

④用户数据报协议 UDP(User Datagram Protocol),提供不可靠的数据传输功能。它不对要发送的数据进行缓冲,直接把从应用层收到的数据加上 UDP 首部发送出去。UDP 不会把应用层多次发送的数据合并成为一个包发送出去。这对于编写简单的请求/响应模式的应用是很方便的,而且利用 UDP 协议对传输速率也没有太大的影响。UDP 协议的数据报格式包括源端口号、目的端口号、数据报长度、UDP 校验和以及 UDP 数据。

⑤上层应用协议,对语音数据打包传输,实现 DSP 与 PC

\* 山东省计算中心 济南 250014

\*\* 山东建筑大学 济南 250014

机的语音数据传输。

## 2 硬件接口电路

硬件接口电路如图 2 所示。

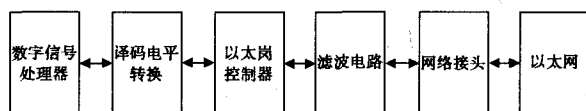


图 2

微处理器采用的是 TI 公司生产的 TMS320DM642 数字信号处理器。它采用了改进的哈佛结构,操作速率达到了 100MIPS。以太网控制芯片采用 RTL8019AS,是台湾 Realtek 公司生产的一种高度集成的以太网控制器。它实现了以太网媒介访问层(MAC)和物理层(PHY)的全部功能。译码、电平变换几项功能由 XC95144xl 完成。它是 Xilinx 公司生产的低功耗 CPLD 器件。HR61001G 是中山汉仁电子有限公司生产的 10BASE-T 接口器件,起低通滤波和隔离变压等作用。

## 3 软件设计

### 3.1 DSP 上嵌入式 TCP/IP 协议的汇编程序设计

由于 DSP 的资源有限,所以网络协议根据嵌入式应用进行了裁减。本设计中完成了 ARP、IP、UDP 等部分协议,既能保证 DSP 接入以太网,又使 DSP 资源能够满足嵌入式系统应用的要求。

首先介绍一下所设计的 DSP 接收和发送以太网数据包所使用的数据结构。作者选择了联合体结构作为 DSP 接收和发送以太网数据包的缓冲区。联合体允许大小和类型不同的定义临时存储在同一存储器空间。这样的好处是各层协议之间数据的传递,实质上是数据指针在传递,而不是数据拷贝传递。而且各层协议可以直接使用该缓冲区,因为缓冲区内的结构体完全是按照各个协议的帧格式设计的。在联合体 databuf 中有四个结构体成员—ethernetpkt、ippkt、udp-pkt、arpkt,分别对应着以太网数据帧、IP 协议、UDP 协议、ARP 协议的帧格式。这 4 个结构体成员是按照它们各自协议的帧格式进行定义的。值得注意的是,因为 databuf 中存储的是 RTL8019AS 发送的以太网数据帧,联合体的成员共用 databuf 缓冲区,所以在各个结构体中要预留低层协议报头存储空间。例如,IP 协议的结构体 ippkt 就要在结构体定义中留下 ethernetpkt 报头的存储空间,即 ippkt 定义中前 9 个整型空间不能使用,从第 10 个整型空间开始才是 IP 数据报的内容。

下面分 4 层介绍所完成的嵌入式 TCP/IP 协议。

3.1.1 物理层:主要完成以太网控制器 RTL8019AS 的复位,寄存器的初始化,并设置 RTL8019AS 的工作方式、中断响应、DMA 通道接收缓冲区的地址设置等。

初始化程序如下:

```

WRITE 0x21,CR /* 设置命令寄存器,选中第 0 页 */
WRITE 0x4c,PSTART/* 页起始地址寄存器初始化,设定接收缓冲区起始页地址 */
WRITE 0x80,PSTOP /* 页停止地址寄存器初始化,设定接收缓冲区终止页地址 */
WRITE 0x4c,BNRY
/* 页边界地址寄存器初始化,设定从接收缓冲区读取数据的位置,读指针 */
WRITE 0x40,TPSR /* 发送页地址寄存器初始化,设定发送缓冲区的起始页地址 */
WRITE 0xcc,RCR
/* 接收配置寄存器初始化,使用接收缓冲区,仅接收自己地址的数据包、广播地址数据包和多点播地址包,小于 64 字节的包丢弃,校验错的数据包不接收 */
WRITE 0xe0,TCR
/* 传输配置寄存器初始化,启用 CRC 自动生成和自动校验,工作在正常模式 */
WRITE 0xc9,DCR
/* 数据配置寄存器初始化,使用 FIFO 缓存,工作在正常模式,DMA 传输数据 16 位宽 */
WRITE 0x01,IMR /* 中断屏蔽寄存器初始化,使能数据包正确接收中断 */
WRITE 0xff,ISR /* 复位中断状态寄存器 */
WRITE 0x61,CR /* 设置命令寄存器,选中第 1 页 */
WRITE 0x4d,CURR
/* 当前页地址寄存器初始化,指向当前正在写的页的下一页,写指针 */
  
```

其中 WRITE 为宏指令,其定义如下:

```

WRITE . macrodata,reg
ST data, *(net_reg)
PORTW *(net_reg),reg
RPT #(20-1)
NOP
. endm
  
```

### 3.1.2 网络层

主要实现 IP 协议和 ARP 协议。当 DSP 接收到正确的以太网数据包以后,调用 check\_packet 函数处理。如果是 ARP 请求,则发送一个 ARP 应答;如果是 ARP 应答,则把对方的 IP 地址和以太网地址放到 ARP 缓存中;如果是 IP 包,则调用 IP 处理模块处理,接收数据。

check\_packet 函数如下:

```

arp: CMPM *(databuf, ethernetpkt.protocol), #0x0608;
判断协议类型是否为 ARP 协议
  
```

BC ip, NTC

CALL arp\_process; 调用 ARP 协议处理模块

ST #taskfree, \*(task); 设置空闲任务指针

B taskfree; 跳转到空闲任务

Ip: CMPM \*(databuf. ethernet. protocol), #0x0008; 判断协议类型是否为 IP 协议

CC ip\_process, TC; 若为 IP 协议, 调用 IP 协议处理模块

ST #taskfree, \*(task); 设置空闲任务指针

B taskfree; 跳转到空闲任务

Arp\_process: 接收 ARP 请求后, 封装好 ARP 数据报, 调用 send\_packet 函数完成发送。在本设计中, DSP 工作复位时, ARP 协议发送一个 ARP 请求, 发送自己的 IP 地址和物理地址给 PC 机。

Ip\_process: 接收 IP 数据报后, 判断协议, 协议如果为 17, 即是 UDP 协议, 则调用函数 udp\_process, 处理 UDP 数据报, 否则丢弃该数据报。在本设计中, 正确接收一帧数据报后, 会给 PC 机一个应答信息, 通知 PC 机已经正确接收了数据报, 可以继续发送数据。应答信息的封装首先调用 create\_udp\_packet 完成对应答信息的 UDP 协议的封装, 然后调用 create\_ip\_packet 函数, 这个函数完成了把封装好的 UDP 数据报再封装成 IP 数据报。最后调用 send\_packet 函数完成发送。

### 3.1.3 传输层

如果 IP 数据报中的协议类型为 17, 则为 UDP 数据报, 需要调用 udp\_process 函数处理数据报。正确接收了数据后, 给发送一个应答信息, 上面已经提到了具体过程, 这里不再复述了。这里介绍一下 UDP 协议的一个难点—UDP 数据的和校验。UDP 和校验要求加上 UDP 伪首部以后再进行校验, 伪首部格式包括 32 位源 IP 地址、32 位目的 IP 地址、8 位数据 0、8 位协议和 16 位 UDP 数据长度。作者根据接收缓冲区的内存管理特点, 即接收缓冲区的成员结构共用同一块数据存储区, 所以将 IP 协议的帧头从生存时间开始到帧头结束封装成 UDP 伪首部。其实这样做, 只需将 8 位生存时间改为全 0, 16 位首部和改为 UDP 数据长度即可, 8 位的协议、32 位源 IP 地址和 32 位的目的 IP 地址不用改动。作 UDP 和校验的时候, 从 IP 协议帧头的报文生存时间开始到 IP 数据报的数据部分结束作带进位的相加, 最后比较对和校验就可以了。

### 3.1.4 应用层

由于传输的是语音数据, 将采集的数据经过 A 律压缩放到 DSP 数据缓冲区 databuf 中, 应用 TCP/IP 协议将其发送给 PC 机。将 PC 机发送来的语音数据 A 律压缩通过 DSP 的多通道缓冲串行口发送给声卡。对于语音数据的处理, 交给 PC 机完成, 这里只实现数据的传输。

## 4 网络控制初始化流程

由于 TCP/IP 栈需要操作系统完成其初始化, 因此必须硬件诊断完成且 DSP/BIOS 初始化完成之后对网络控制进行初始化。

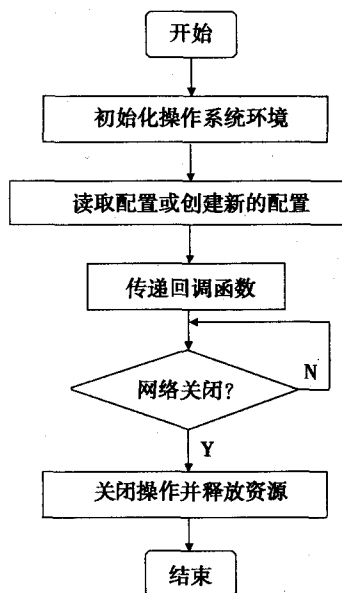


图 3

在调用 TCP/IP 协议相关函数前, 先初始化操作系统环境, 再创建一个新的配置, 或从只读存储器中导出一个已存在的配置, 然后把配置句柄及回调函数 start、stop 和 IP 地址变换操作传递给协议栈。直到网络关闭时, 应用程序关闭在 start 中调用的操作并释放分配的资源, 退出网络协议栈, 网络控制初始化流程如图 3 所示。

## 5 结束语

本文在 DM642 芯片上实现了一个高度精简的嵌入式 TCP/IP 协议栈, 论述了协议栈的硬件接口设计和软件实现。可以方便地在 DSP 上实现网络视频的传输, 本协议栈目前已成功应用于数字视频监控系统中, 运行状况良好, 该协议栈的实现对于数字视频广播系统、视频点播系统等实时多媒体处理的应用也有借鉴作用。

## 参考文献:

- [1] 江思敏, 刘畅. TMS320C6000 DSP 应用开发教程, 北京: 机械工业出版社, 2005.
- [2] 陶风, 罗白玲. DSP 理论基础与工程实现, 北京: 国防工业出版社, 2006.
- [3] 胡道元. 计算机网络指南, 北京: 电子工业出版社, 2003.
- [4] 潘爱民. 计算机网络, 北京: 清华大学出版社, 2005.
- [5] 刘谦, 苏建平. TCP/IP 透彻理解, 北京: 电子工业出版社, 2004.