

# 基于 DM642 DSP 的网络化数据采集与编程实现

杨宇晓, 宋茂忠, 苗雄峰

(南京航空航天大学 信息科学与技术学院, 江苏 南京 210016)

**摘要:**随着数字信号处理器性能的不断提高和网络技术的飞速发展,将采集到的数据通过网络实时传递给需要的用户已成为可能。提出了一种基于 DSP 的网络化数据采集系统设计方案。利用 DM642 芯片的多通道缓冲串口 (McBSP) 和 EDMA 控制器实现数据采集和搬移,为保证采集数据的完整,EDMA 通道采用 ping-pong 缓冲技术以实现不间断地接收数据源的输出信号。基于实时性考虑,本系统采用 UDP 数据报传输协议,将数据通过 EMAC 和 RJ-45 端口发送至网络。这种方式实现简单、干扰少、传输距离可扩展,可使网络中的用户实时共享采集到的数据。

**关键词:**DM642;数据采集;TCP/IP

**中图分类号:**TP393

**文献标识码:**A

**文章编号:**1673-629X(2008)06-0188-04

## Networked Data Collection System Based on DM642 and Programming Approach Accomplishment

YANG Yu-xiao, SONG Mao-zhong, MIAO Xiong-feng

(College of Information Science & Technology, Nanjing University of  
Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract:** With the constant improvement of digital signal processor performance and high-speed development of network technology, it becomes possible to transmit collected data to users on real time by network. The following thesis put forward a new design that network data collection system based on DSP. The data collection and data transference can be accomplished by means of McBSP (multi-channel buffer series port) of DM642 chip and EDMA controller. EDMA channel adopt ping-pong buffer technology, so as to guarantee the collected data intact, and realize continuously receiving output signals from the data base. Take the real time into consideration, this system adopt UDP (user datagram protocol) transmitted the data to network by EMAC and RJ-45 terminal. The design is easy accomplishment, less-interference, and also has expandable transmission distance. So the users on the Internet can share the collected data on real time.

**Key words:** DM642; data acquisition; TCP/IP

## 0 引言

近些年来,随着数字信号处理器性能的不断提高,DSP 广泛应用于通信控制、信号处理、仪表仪器、医疗、军事、消费类产品、家电及工业检测等领域。DSP 芯片的快速数据采集和处理能力以及片上集成的各种功能模块使 DSP 能够应用于各种场合,其中数据采集是 DSP 最基本的应用领域。另外随着互联网技术的飞速发展,将采集到的数据通过网络实时传递给需要数据

的用户也成为可能,文中提出了一种利用 TI 公司的 DM642 DSP 芯片来实现网络化数据采集系统的设计方案。

## 1 开发平台的介绍

TI 公司的 DM642 是一款强大的多媒体处理器。该 DSP 内部时钟高达 600MHz,8 个并行运算单元,最大处理能力达到 4800MIPS,外部总线时钟 100MHz。DM642 在 C64x 的基础上增强了外围设备和接口,主要包括:3 个高精度、可配置的视频端口;遵循 IEEE802.3 协议,具有媒体独立接口——MII (media-independent interface),独立发送和接收通道 10/100Mb/s 以太网控制器 (EMAC);多通道音频串行端口 (McASP);两个多通道缓冲串口 (McBSP);64 个通道的增强型 DMA 接口;用户可配置的 16 位或 32 位的

收稿日期:2007-09-06

基金项目:国家自然科学基金资助项目(60572108);航空科学基金(20060152003)

作者简介:杨宇晓(1984-),男,河南洛阳人,硕士研究生,研究方向为通信与信息系统;宋茂忠,教授,研究方向为无线通信与无线定位。

主端口接口(HPI16/HPI32);64 位的外部存储单元接口<sup>[1]</sup>。它的完全可编程性又可以使得它能够兼容正在发展的各种多媒体信号处理标准,构成通用的多媒体通信系统软件平台。这些特性使得它在数字视频处理领域具备了广泛应用的条件。

瑞泰创新科技有限责任公司推出的 ICETEK - DM642 - PCI 开发板是一个基于 DM642 芯片的低功耗独立的开发平台,用户可以对 TI 的 DM642 DSP 进行评测和开发应用。板上有一个 600MHz 的 DM642 芯片,还集成了 4MB Flash, 32MB 100MHz 的 SDRAM,另外还有音视频采集和输出端口,10/100M 以太网接口,仿真器的 JTAG 接口。我们的网络化数据采集系统就是在这块板上实现的。

## 2 系统总体设计

基于 DSP 的数据采集是利用 TI 公司的 DM642 处理器的多通道缓冲串口(McBSP)来接收数据的。这种多通道缓冲串口是一种高速的多通道的同步串口,具有速率高和同步串行传输特性。通过它将需要的数据采集到 DSP 中,经过 DSP 处理后最终由 EMAC 发向网络。系统原理框图如图 1 所示。

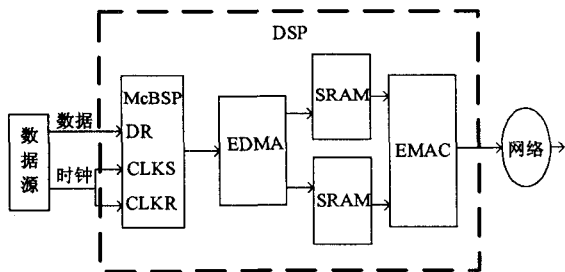


图 1 系统原理框图

### 2.1 数据采集模块

数据采集模块利用了 DM642 开发板 ICETEK - DM642 - PCI 上的背板 2(DP\_C2)将数据源信号引入到评估板中,由 DM642 中的片内外设多通道缓冲串口(McBSP)采集到其 DRR 中,通过 EDMA 控制器将 DRR 中的数据搬移到合适的缓存中等待 CPU 进行处理。

多通道缓冲串口(McBSP)接口是由与外部器件相连接的一个数据通道和一个控制通道组成。不同的接收和发送引脚实现和外部器件之间的通信。McBSP 的接收操作采用三级缓冲方式。接收的数据到达 DR 引脚后移位到接收转移寄存器(RSR),一旦整个数据单元接收完毕,如果接收缓冲寄存器(RBR)未满,则 RSR 被复制到 RBR 中,最后 RBR 被复制到数据接收寄存器(DRR)中,CPU 或 EDMA 控制器就可以从中读取数据<sup>[2]</sup>。

本系统是由 EDMA 控制器将 McBSP 的 DRR 寄存器数据搬移到缓存中,等待后续处理。DM642 有 64 个 EDMA 通道,利用 EDMA,可以实现片内存储器(L2 SRAM)、片内外设以及外部存储空间之间的数据转移。EDMA 搬移数据时不受 CPU 的控制,在这期间 CPU 可以完成其它操作。McBSP 和 EDMA 通过 McBSP 的 REVT 事件来实现之间的同步,McBSP 在 DRR 寄存器被准备好之后,就会向 EDMA 控制器发送 REVT 事件,通知 EDMA 来读取数据。EDMA 控制器的 15 号通道则处理来自多通道缓冲串口(McBSP)的 REVT 事件。在系统中 EDMA 控制器的通道 15 的数据源地址则定为 McBSP 的 DRR。

为了能够不间断地接收数据源输出的信号,EDMA 通道分配有两个存储器空间作为其数据目的的空间,一个为 Ping 空间一个为 Pong 空间,交替为 EDMA 通道服务。利用 EDMA 的连接功能使得当 EDMA 通道将 Ping 空间填满时,其目的数据空间变为 Pong 空间;当 EDMA 通道将 Pong 空间填满时,其目的数据空间变为 Ping 空间,从而实现在 CPU 处理数据时还可以连续不断地接收数据到缓存中,而不会产生资源冲突。当 EDMA 控制器将一个数据空间填满之后会产生 EDMA\_INT CPU 中断通知 CPU 处理相应的存储空间中的数据。相应的数据采集模块功能框图如图 2 所示。

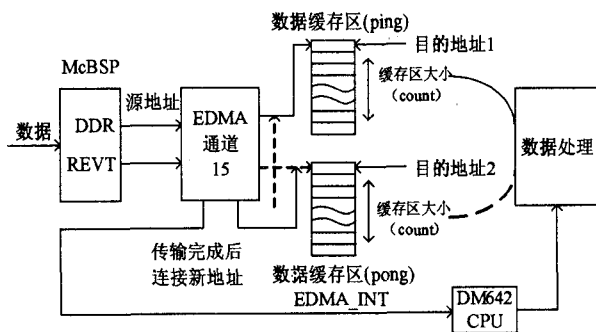


图 2 数据采集模块功能框图

### 2.2 网络功能模块

以太媒介访问控制器(EMAC)控制着从 DSP 到物理层(PHY)的包数据流,管理数据输入输出模块(MDIO)控制着 PHY 的配置和状态的监视。EMAC 在 DSP 核心处理器和网络之间提供一个高效接口。EMAC 和 MDIO 模块通过一个普通接口与 DSP 连接,这个普通接口被称作 EMAC 控制模块,也可以被认为是 DSP 的 EMAC/MIDO 外设<sup>[3]</sup>。EMAC 控制模块框图如图 3 所示。

TI 公司为 TMS320C5000TM 系列和 TMS320C6000TM 系列数字信号处理芯片(DSPs)设计了一种

嵌入式实时操作系统内核 DSP/BIOS, 可以使开发者能够开发出性能优良的程序而不必拘泥于实时性的限制, 但是它不包含网络功能。为了解决这个问题, TI 推出了 TCP/IP NDK (Network Developer's Kit)。该开发包采用紧凑的设计方法, 实现了用较少的资源耗费支持 TCP/IP NDK, 仅用 200~250kB 程序空间和 95kB 数据空间即可支持常规的 TCP/IP 服务, 包括应用层的 Telnet, DHCP, HTTP 等。所以, NDK 很适合目前嵌入式系统的硬件环境, 是实现 DSP 上网的重要支撑工具<sup>[4]</sup>。

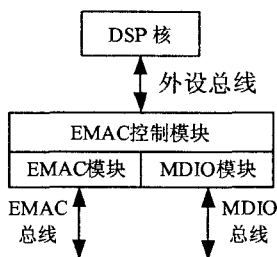


图 3 EMAC 控制模块框图

与常规的 TCP/IP 开发不同, 在开发嵌入式应用时, 必须对网络环境和应用需求做更细致的设置。利用 NDK 可以迅速在 DSP 应用上集成 TCP/IP 协议栈, 大大加快开发进程。

### 3 系统软件设计

根据设计系统要求, 要能够实时接收数据, 并将接收到的数据实时地发送到网络中, 因此整个系统的软件设计主要分为两部分, 数据采集部分和网络部分。数据采集部分则主要完成多通道缓冲串口 McBSP 和 EDMA 的配置、各种中断的管理和采集到的数据处理。网络部分主要包括系统 TCP/IP 协议栈的配置、HTTP 和 DHCP 等网络服务的设计、系统网络配置的管理以及对采集到的数据进行网络发送等。此外为了开发板工作稳定还需要一定的系统初始化。系统软件设计的总框图如图 4 所示。

#### 3.1 系统数据采集软件

DSP/BIOS 是一个简单的嵌入式实时操作系统, 它大大方便了用户编写多任务应用程序, 而且 DSP/BIOS 是高度集成的, 只需大约几 k 大小的程序空间。用户可以随时通过 API 接口 (C 语言或是汇编语言) 唤醒内核服务程序。此外, DSP/BIOS 还提供了一个图形配置工具 (Graphical Configuration Tool), 开发者只需简单选择自己需要的服务内核加入自己工程即可<sup>[5]</sup>。静态配置和动态生成操作系统对象代码, 大大减少目标系统内存开销, 简化程序设计, 降低系统复杂度。

系统数据采集部分设计主要包括中断、McBSP 和 EDMA 控制器的配置。在对于中断、McBSP 和 EDMA 控制器的配置时按照 2.1 节数据采集模块的设计要求, 充分利用 DSP/BIOS 的 GUI 工具和 CSL (Chip Support Library) 以加快系统开发的进度。

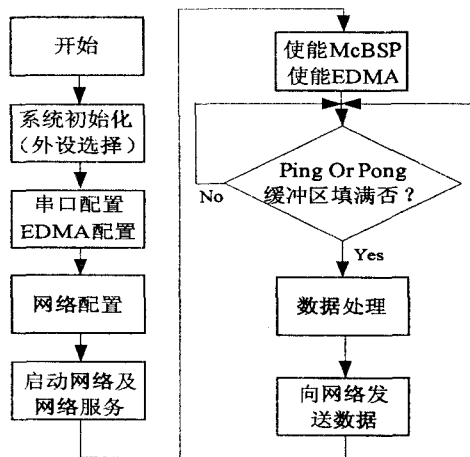


图 4 系统软件设计总图

#### 3.2 系统网络模块软件

系统的网络部分软件主要负责系统 TCP/IP 协议栈的配置、启动、管理以及采集数据的网络发送。本系统用 UDP 数据报实现数据的网络发送, UDP 是一个无连接协议, 传输数据之前服务器端和客户端不建立连接, 较适合实时网络通信<sup>[6]</sup>。

由于 NDK 已经提供了完整的 TCP/IP 库函数, 代码只需按需要进行配置即可。采集数据的网络发送模块, 在 CPU 响应 EDMA\_INT 中断的中断服务子程序中实现。

在采集数据的网络发送程序中首先读取中断挂起寄存器 CIPR 的值判断哪个存储空间被充满 (Ping 或者 Pong), 当 ciprl = 0x0010 时, 表示 EDMA 完成 ping 存储空间的传输, ciprl = 0x0040 表示 EDMA 完成 pong 存储空间的传输。然后对完成传输的存储空间数据进行组包, 完成后调用 UDP 发送函数, 将组好的数据包发送出去, 核心代码如下:

```

#define PINGPONGBUFSIZE 3072
#define SENDLENGTH 256
long Sorstart = 0;
//EDMA_INT 中断服务子程序
void EdmaTccIsr(void)
{
    ciprl = EDMA_RGET(CIPRL);
    if((ciprl & 0x0010))
    { //寄存器 CIPRL 清零
        EDMA_RSET(CIPRL, 0x0030);
        for(i=0; i<12; i++) //数据组包、发送
  
```

```

    dataPack(pingRx);
    UdpClt();
}
else if((ciprl&0x0040))
{ // 寄存器 CIPRL 清零
    EDMA_RSET(CIPRL,0x00C0);
    for(i=0;i<12;i++)//数据组包、发送
    { dataPack(pongRx);
      UdpClt();
    }
}
//数据组包函数
void dataPack(Uint32 inData[])
{
    for(i=0;i< SENDLENGTH;i++)
    {
        sendUDP[i] = inData[Sorstart+i];
        //主机字节顺序转换成网络字节顺序
        sendUDP[i] = htonl(sendUDP[i]);
        Sorstart = Sorstart + SENDLENGTH;
        Sorstart = Sorstart % PINGPONGBUFFSIZE;
    }
}
void UdpClt(void)//网络发送函数
{ //创建 socket
    SOCKET sockClt;
    .....
    //设定目的地址端口属性
    bzero(&ClientAddr,sizeof(struct sockaddr_in));
    ClientAddr.sin_family = AF_INET;
    ClientAddr.sin_addr.s_addr = IPAddr;//发送地址
    ClientAddr.sin_port = htons(7001);//发送端口号
    .....
}

```

(上接第 187 页)

## 5 结束语

文中在对军队报务训练功能需求分析的基础上,详细论述了基于 Intranet 的无线电报务训练平台的系统架构和实现方案,并对其主要应用的关键技术进行了阐述。该系统为军队报务训练提供了一个良好的训练平台,能依托军内互联网代替实装开展报务训练,有效地提高军队报务人员的训练水平,促进军队通信整体作战能力的提高。

### 参考文献:

- [1] 张 勇,赵东宁,江光杰.军事短波通信抗干扰性能仿真设计与实现[J].系统仿真学报,2003,30(1):17-20.

```

//发送数据
sendto(sockClt, sendUDP, (int) sizeof(sendUDP), 0, (PSA)
&ClientAddr, sizeof(struct sockaddr));
.....
}

```

在客户端 PC 上运行相应的 UDP 接收端程序,即可实现通信。若将此系统连接至路由器,并进行相应配置,则可以实现采集数据的网络共享。

## 4 结束语

本系统利用 DM642 的多通道缓冲串口 McBSP 和 EDMA 实现对数据源信号的采集,利用 DM642 的 EMAC 和 RJ-45 端口完成了采集数据的网络发送,较好地实现了网络化数据采集系统的功能。

### 参考文献:

- [1] Texas Instruments. TMS320C6000 DSP Peripherals Overview Reference Guide[R]. [s.l.]:[s.n.],2005.
- [2] Texas Instruments. TMS320C6000 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide[R]. [s.l.]:[s.n.],2004.
- [3] Texas Instruments. TMS320C6000 DSP EMAC and MDIO Module Reference Guide[R]. [s.l.]:[s.n.],2005.
- [4] Texas Instruments. TMS320C6000 TCP/IP Network Developer's Kit NDK User's Guide[R]. [s.l.]:[s.n.],2001.
- [5] Texas Instruments. TMS320C6000 DSP/BIOS Application Programming Interface(API)[R]. [s.l.]:[s.n.],2005.
- [6] 杜 文,沈 勇,唐 昆.DM642 上 TCP/IP 协议的实现及性能测试[J].微计算机信息(嵌入式与 SOC),2006,22(5-2):149-152.

- [2] Levine P, Gauger J R D, Bwera L D, et al. A comparison of mouth stick and Morse code text inputs[J]. AAC augmentative and alternative Communication, 1996, 51(2):45-51.
- [3] 宋继红,杨 放,郝永庆.基于 winsock 的多线程网络通信开发[J].辽宁大学学报:自然科学版,2003,30(2):167-169.
- [4] 刘玮玮. Visual C++ 视频/音频开发实用工程案例精选[M].北京:人民邮电出版社,2005:395-424.
- [5] Hughes C, Hughes T. Object-Oriented Multithreading Using C++[M].北京:人民邮电出版社,2003:56-89.
- [6] 乔 林,杨志刚. Visual C++ 6.0 高级编程技术——MFC 与多线程篇[M].北京:中国铁道出版社,2000:137-216.
- [7] 阎 宏. JAVA 与模式[M].北京:电子工业出版社,2002:179-197.