

基于 NDK 的数字电视传输流(TS)网络采集系统

电视发展到今天已经进入数字化的时代了，数字电视系统取代模拟电视系统的一大优势在于实现了信息的数字化。电视节目信息实现数字化，所有的信息可以在数字世界里传播、转换、存储和处理，从而使得广播电视网能够和计算机网络相连接，大大提高了信息处理的能力。

概述

TCP/IP 是 Internet 网络上传输数据所必需的协议，这种网络通信模式在 PC 机之间的实现已经完善，但是由于体积、价格等因素限制了应用的范围。因此，基于 TCP/IP 协议与 Ethernet 的嵌入式系统网络通信设计成为目前一个热门的话题。

本系统实现了在以 TI 公司的 TMS320DM643 为核心的嵌入式系统中对数字电视传输流信号进行采集与在 Ethernet 中传输。利用本系统可轻松的实现在局域网中对数字电视传输流信号的传输、调度。数字电视传输流信号源是针对欧洲数字有线广播系统标准(DVB-C)的数字有线电视信号。网络接入硬件是在以 TI 公司的 TMS320DM643 为核心的嵌入式系统中实现，网络接入软件采用了 TI 公司的针对 C6000 系列 DSP 推出的 TCP/IP NDK(Network Developer's Kit)网络开发包来实现。

系统的电路设计

电路部分主要由五部分组成。数字电视传输流网络采集系统框图如图 1 所示。

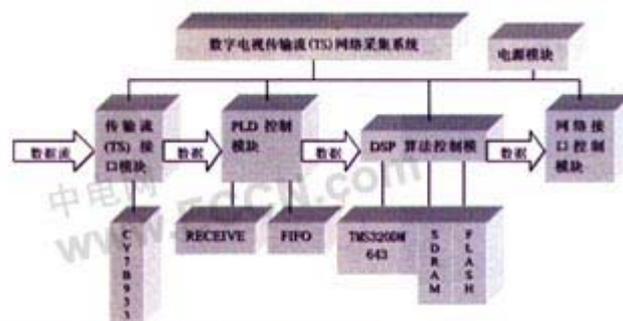


图 1 数字电视传输流(TS)网络采集系统框图

主要功能是通过传输流接口模块采集数字电视信号进入 PLD (CYCLONE EP1C6Q24 0C8) 芯片，进行必要的处理后，将信号发送到 DSP(TMS320DM643)芯片储存起来并进行算法处理。通过 TMS320DM643 芯片对 BCM5221 芯片进行必要的配置，将储存在 TMS320DM643 芯片内的数据通过 BCM5221 芯片传送到局域网中，并通过计算机接收数据。

传输流(TS)接口模块

传输流接口模块由 CY7B933 输入接口芯片及其电器接口电路组成。CY7B933 输入接口电路是点对点的传输模块，可以通过光纤、同轴电缆和双绞线进行高速的串行数据传输。输入接口符合 DVB-ASI 的接口标准。输入接口接收到串行位流后，通过内部 PLL 时钟同步恢复数据的时钟信息，并对位流进行串并转换、解码和传输检错等操作。这种输入接口能灵活的实现把高速点对点串行数据转变成并行数据，而且应用领域广泛，包括各种服务器，存储器 and 视频传输的应用。

PLD 控制模块

在项目中，此部分硬件选用的是 Altera 公司的 CYCLONE EP1C6Q240C8 芯片。

此模块主要的功能是实现与 CY7B933 接收芯片的接口，把数据从 CY7B933 接收进来，并缓存数据。这部分功能均由 VHDL 语言编写的功能模块实现。主要有两个功能模块：

RECEIVE 模块与 FIFO 模块。RECEIVE 模块主要负责从 CY7B933 接收芯片接收数据字段进来；FIFO 模块主要负责缓存数据。功能框图如图 2 所示，下面分别对 RECEIVE 模块与 FIFO 模块的功能进行介绍。

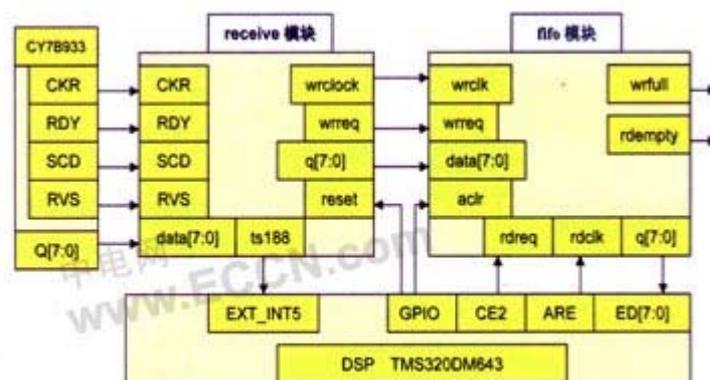


图 2 PLD 控制模块

RECEIVE 模块

RECEIVE 模块的功能是实现与 CY7B933 接收芯片的接口，把数据从 CY7B933 接收进来。其工作方式是以一个传输流包为边界接收数据的。

首先 RECEIVE 模块会检测传输流包的边界，通过查找包头字节（固定为 0x47）间的字节数来确定。因为包中数据也可能含有 0x47，所以要牺牲三个包的数据来检查三次。当发现 0x47 这个字节的时候，就会触发一个内部的计数器开始计数。当计数到 188 后，如果下一个字节又是 0x47，说明传输流包属于 188 个字节的包，那么计数器被清零；如果下一个字节又是 0x47，说明传输流包属于 188 个字节的包，那么计数器被清零，否则计数器清零并重新开始检测边界。

当检测到边界以后，RECEIVE 模块开始接收数据包。计数器会从零开始计数，在接收数据的过程中使能 wrreq 输出有效信号，同时把数据输出到下一级。当计数到 188，表示一个数据包接收完成。当一个包的数据接收完之后，计数器清零，并置 ts188，保持高电平一个时钟周期。下一个周期检测数据是否为 0x47，如果是说明是下一个数据包的边界，否则说明出现了错误，并重新回到上一段所说的检测数据包边界的状态。

此外，PLD 模块内会有一个专用计数器记录空包数，当接收到数据包后会首先检测此数据包是否空包，如果是空包 PLD 模块会把这个空包删除，并在计数器中加一。如果接收的不是空包就会把计数器的值加到这个数据包的私有字段中，并缓存到 FIFO。然后计数器自动清零。这样处理数据包的目的是为了减少网络传输的数据流量，从而可以传输更多的传输流数据。把计数器的值加入私有字段是为了在计算机接收到数据后可以把原来的空包恢复出来，从而保证原传输流数据的完整性。

fifo 模块

fifo 模块的功能实现从 RECEIVE 模块接收数据，并缓存起来。当 RECEIVE 模块接收完一个完整的传输流的数据包之后，会发送 ts188 或 ts204 的中断信号给 DSP，DSP 就会启动 EDMA 功能从 fifo 模块接收数据。DSP 与 fifo 模块是采用异步连接的方式，具体的接收操作在 DSP 部分说明中加以描述。

DSP(TMS320DM643)算法处理模块

此模块主要以 TI 公司的 TMS320DM643 为核心的嵌入式系统组成。主要实现从 PLD 模块接收传输流数据包，把数据包打包成 TCP/IP 格式，并实现对网络接口(BCM5221)控制模块的初始化，然后把数据包传送到网络模块。

为了实现上述功能，必须建立起一套以 TMS320DM643 为核心的基本系统。系统各个组成部分如图 3 所示。

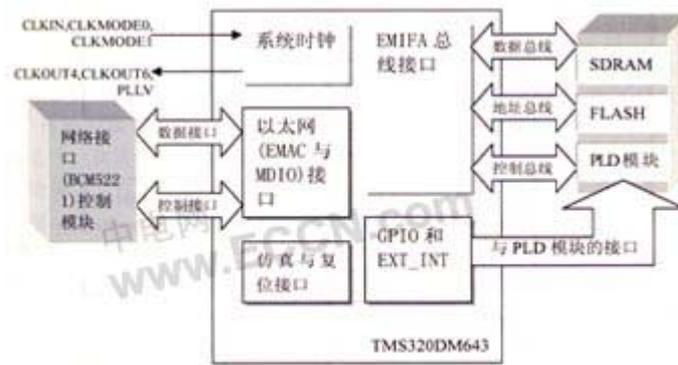


图 3 TMS320DM643 系统框图

系统的具体配置

时钟配置: EMIF 内核时钟 ECLKIN 是 133MHz。此外，系统上外设总线、EDMA 传输和 L2 存储器的工作时钟为 CPU 内核时钟的 1/2，即 300MHz；片上定时器的工作时钟为 CPU 内核时钟的 1/8，即 75MHz。

中断配置: TMS320DM643 除了 RESET 和 NMI 引脚提供外部不可屏蔽中断请求输入以外，还有 2 个外部中断引脚 GP0[5]/EXT_INT5、GP0[7]/EXT_INT7 提供可屏蔽的外部中断请求输入。系统中 EXT_INT5 外部中断用作 PLD 模块的请求接收数据信号，每当 PLD 模块接收完一个传输流(TS)包就会发送一个外部中断信号给 DSP，通知 DSP 接收数据。此外，EDMA 中断用于当接收完一个包的数据后做后续处理。

系统对 EMIF 的使用情况:

系统在 CE0 空间扩展了 4 Mx64-bit 的 S DRAM 存储器(MT48LC4M32BPG)，用于存储程序与数据。SDRAM 的工作时钟由 TMS320DM643 的 E CLKOUT1 提供，与 EMIF 的工作时钟频率相同，本系统中默认为 E CLKIN 为其时钟源，即 133MHz。

系统在 CE1 空间扩展了 4Mx8-bit 的 Flash 存储器(Am29LV033C)。在对 FLASH 进行读/写访问前，需要通过 EMIF 的 CE1 控制寄存器 CE1CTL 将 CE1 空间配置为 8-bit 异步存储器接口，及读/写时序。

系统在 CE2 空间扩展了与 PLD 模块的 fifo 模块连接的接口。在 DSP 看来，fifo 模块可视为 8-bit 异步只读存储器。如图 2 所示，fifo 模块的读使能信号 rdreq 与 TMS320DM643 的 CE2 片选信号连接；fifo 模块的读时钟信号 rdclk，与 TMS320DM643 的 ARE 读使能信号连接。

以太网(EMAC 与 MDIO)接口

TMS320DM643 上集成有一个 EMAC+MDIO 片上外设，EMAC 是 Ethernet Media Access Controller 的缩写，即以太网媒体访问控制器，MDIO 是 Management Data Input/Output 的缩写，即管理数据输入/输出模块。EMAC+MDIO 用于为以太网物理层(PHY)器件提供接口，其中 EMAC 为接口以太网 PHY 提供数据通路，MDIO 为接口以太网 PHY 提供管理信息通路。以太网接口主要功能如下：符合 IEEE 802.3 协议；支持传媒无关接口(MII)；八个独立的发送与接收通路；同步的 10/100Mbit 的数据操作；广播及多帧的传送。

TMS320DM643 的网络接口的原理框图如图 4 所示。

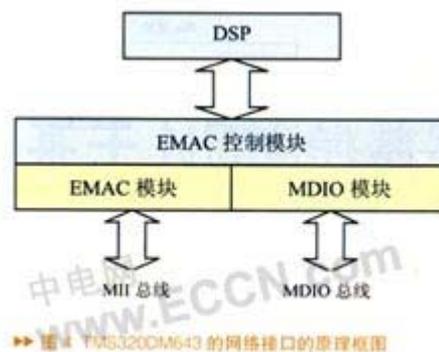


图 4 描述了 EMAC+MDIO 与 DSP 中间有一个 EMAC control module 模块。它主要包含一些必备的使 EMAC 更加有效的使用 DSP 的存储空间，控制其复位、中断的一些逻辑。这些寄存器的地址空间为：0x1C800000~0x1C803FFF。

网络接口(BCM5221)控制模块

本系统中用 Broadcom 公司的 BCM5221 作为 10/100Base-TX 以太网收发器，BCM5221 的 MII 接口与 TMS320DM643 的 MII 接口对接。具体接口如图 5 所示。RJ45 连接器选用 AMP 公司的 406549-1，其上带两个 LED 指示灯，右边的 LED 为绿色，用作指示连接状态。左边的为黄色，正常情况下用来指示数据传输。

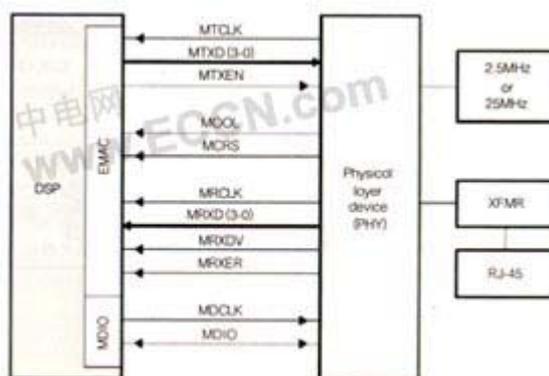


图 5 TMS320DM643 与 PHY 设备连接的框图

电源模块

系统包括四组电源：系统外接稳压电源，把 220V 的交流电源电压转换成 5V 直流电压；PLD 模块电源有两种电源供电，分别是 3.3V 和 1.5V；TMS320DM643 需要 2 种电源，分别为 CPU 核心和周边的 I/O 接口供电。周边 I/O 电压要求 3.3V，CPU 核心电压只要 1.4V；网络接口控制模块采用 3.3V 电源供电。

系统的软件设计

系统的软件设计主要是通过 TI 公司推出的专门针对 C6000 系列的 TCP/IP 网络开发包 NDK(Network Developer's Kit)来实现的。该开发包支持 TCP/IP 协议，并占用较少的系统资源。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。所以，NDK 很适合目前嵌入式系统的硬件环境，是实现 DSP 网络开发的重要工具。

系统软件的框图如图 6 所示。

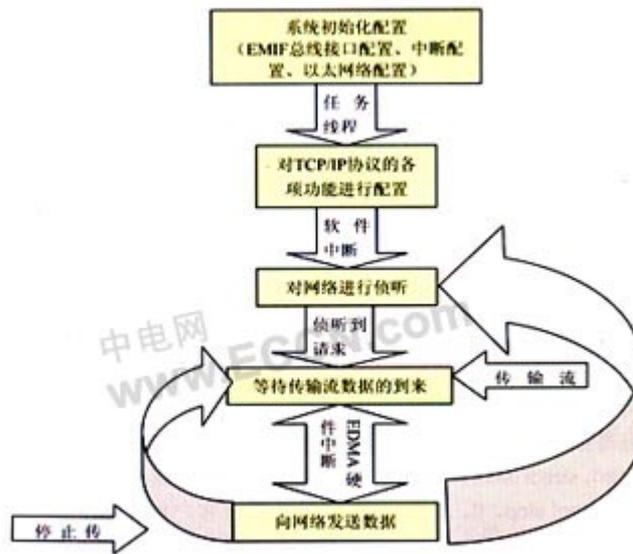


图 6 软件系统框图

具体流程：系统配置主要在初始化函数中完成，此初始化函数将在.cinit 初始化后，并在 DSP/BIOS 初始化和 main 函数之前调用。系统配置需要开启 INT8 中断，也就是 EDMA 传输中断，并设置 GPIO7 引脚为高电平，从而使 PLD 模块始终处于初始状态。TCP/IP 协议配置在任务线程中完成，主要实现服务器 IP 地址和网关的配置。当其完成后，系统就会触发一个软件中断，在软件中断程序中，服务器会产生一个用于侦听的端口，并开始侦听网络上的请求。此时，服务器已经处于待命状态，等待客户机的传输请求。

当客户机的传输请求到来时，系统首先会使 GPIO7 引脚变为低电平，PLD 模块开始发送数据。当其接收到一个完整的传输流数据包后就会发一个中断给 DSP，DSP 启动 EDMA，用 ping-pong 的方式接收数据进入特定的存储器，当接收完一个完整的传输流数据包后，会产生硬件 EDMA 中断，中断程序主要是设置特定的标志为 1，然后退出硬件中断，重新回到软件中断程序。当软件中断程序检测到特定的标志为 1 时，就会开始向网络中传输在存储器中的传输流数据包给客户机。当一个数据包传输完毕后，特定的标志会被置 0，系统等待下一个数据包的到来。

当客户机要求停止传输的请求到来时，系统会重新使 GPIO 引脚变为高电平，PLD 模块回到初试状态。此时系统也会处于待命状态，继续侦听网络。

系统初始化

初始化配置主要包括了 EMIF 总线配置、中断配置和底层网络配置三个部分。初始化工作首先在 DSP/BIOS 的全局参数配置窗口进行设置，然后调用初始化函数进行配置。

在 DSP/BIOS 的全局配置窗口主要设置了 DSP 的工作时钟频率为 600MHz，选择使用的片级支持库 CSL 为 DM642 的库，选用小端访问模式，片内 256k 的 SRAM 全部用于 CACHE 高速缓存，并调用初始化函数 dm643_init()。

TCP/IP 协议配置

初始化程序 dm643_init()完成后，从现在起系统将进入各个线程。首先，系统会触发任务线程 TSK0，在 TSK0 任务线程中执行函数 StackTest()，其首先调用 NC- SystemOpen()函数完成协议栈及其所需内存的初始化，然后新建一个系统配置句柄 hCfg = CfgNew()，并实现服务器 IP 地址和网关的配置。在本系统中，IP 地址配置为“192.168.0.2”；子网掩码为“255.255.255.0”；网关地址为“192.168.0.1”。配置好之后，系统会触发一个软件中断，并做好侦听网络的准备。

网络侦听

前面在任务线程中已经配置了 IP 地址和网关，下面就要在软件中断中设置端口和进行侦听。首先在开始使用报路之前，必须分配一个文件环境 fdOpenSession(TaskSelf()) 给这个报路。接着就可以创建一个报路对象 stcp = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP)，并设置端口 sin1.sin_port = htons(1000)，在这里我们设置端口号为 1000，当然也可以设置其他的端口号。然后把端口号与报路对象绑定 bind(stcp, (PSA) &sin1, sizeof(sin1)。最后进入侦听状态 listen(stcp, 1)。

数据的接收

当客户端要求传输数据的信号到来后，系统会使 GPIO7 引脚变为低电平，PLD 模块开始发送数据。DSP 通过 EDMA 方式来接收数据。EDMA 可以在没有 CPU 参与的情况下，由 EDMA 控制器完成 DSP 存储空间内的数据搬移。系统主要采用 EDMA 的 ping-pong 方式来连续接收数据。用 ping-pong 的方式的目的是为了使接收操作和发送操作分开处理，争强程序的操作性和可读性。

网络发送

接收完一帧的数据后会触发硬件中断，在硬件中断函数中会把特定的标志置为 1，然后从硬件中断函数中返回，并进入软件中断函数。软件中断函数把刚刚接收到的数据发送到网络中。

软件中断函数会一直侦听是否有新的 TCP/IP 报路连接请求，当检测到有新的报路连接请求，就会创建一个报路来连接到请求连接的应用程序。建立连接后，软件中断函数会等待客户机的命令。客户机此时会发送一个传输命令给服务器，服务器接收到这个命令后，软件中断函数就会准备发送数据。

软件中断函数会等待一帧传输流数据的到来，也就是等待特定的标志置 1。当检测到标志位置 1 后，软件中断函数就会把刚刚接收到的数据包发送出去。发送成功后，又会把标志位置 0，重新等待下一个数据包的到来。如此循环的把数据包发送出去。

当收到结束连接的请求后，会清楚标志位，并设置 GPIO7 引脚为高电平，从而使 PLD 模块回到初始状态。然后关闭连接，并回到侦听状态。

结束语

本系统的硬件和软件功能已经实现，网络传输的速度为 2Mbytes/s。按照此速度，假设传输流中的空包百分比为 40% 计算，可以传输 5 Mbytes/s 的传输流数据，也就是 50Mbit/s 的传输流，至少可以传输 2~3 路的 TS 流节目。因此，可以证明本系统的设计方法合理、有效。

DSP(TMS320DM643)在数字信号处理方面具有的比其他芯片更为强大优势，现在我们已经可以在 DSP(TMS320DM643)系统中实现了数字电视传输流的网络功能，日后可以在本系统中继续增添新的数字信号处理，特别是视音频处理模块，从而实现更强大的功能。