嵌入式 TCP/IP 详解

1.1 TCP/IP 协议介 绍

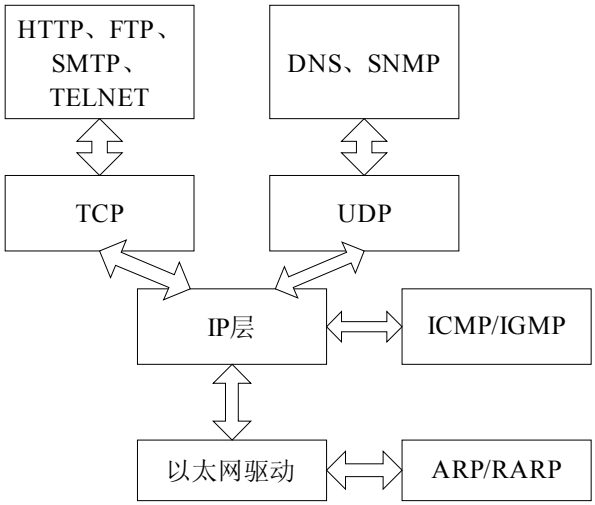

图 1.1 TCP/IP 协议结构图

Fig. 1.1 Structure of TCP/IP protocol

如图 1.1 所示，以上 TCP/IP 协议栈可分为链路层协议、网络层协议、传输层协议、应用层协议四大部分。以下简单介绍这四层协议。

（1）链路层协议

主要有地址解析协议 ARP (Address Resolution Protocol) 和反地址解析协议 RARP (Reverse Address Resolution Protocol) 在 TCP/IP 协议通讯中，涉及到的地址是 IP 地址 (IPv4 为 32bit)，这是来自网络层的地址，然而以太网都有自己的寻址机制(为 48bit 地址)，所以两层之间必须进行地址之间的转换。向以太网中

发送 IP 数据时，首先根据目的 IP 地址在 ARP 高速缓存表中查询相应的以太网地址。如果查到匹配的结点，则相应的以太网地址被写入以太网帧首部，数据报被加入到输出队列等候发送。如果查询失败，ARP 会先保留待发送的 IP 数据报，然后广播一个询问目的主机硬件地址的 ARP 报文，等收到回答后再将 IP 数据报发送出去。RARP 协议恰恰相反，它负责将以太网地址转化为 IP 地址。

(2) 网络层协议

网络层主要涉及到 IP 协议、ICMP 协议和 IGMP 协议。IP 协议是 TCP/IP 中的重点，所有的应用都要通过它在 Internet 进行数据传输，IP 协议提供的是不可靠、无连接的数据分组传送服务。不可靠性是指 IP 协议中不保证 IP 数据分组能成功的到达目的地，但是它提供很好的传输服务，它有一个简单的算法：当发生错误时，它就丢弃数据分组，并发送 ICMP 消息给信息源端。无连接是指 IP 协议中不维护任何关于后续数据分组的信息，数据分组之间是独立的，从源端到目的端的路径也是不相关的。ICMP 协议主要是用于差错控制，所涉及的错误信息包括要有目的地不可达到、报文超时、无效头字段、源端拥塞、路由重定向、回声请求和回声应答、时间标记请求和时间标记应答、地址掩码请求和地址掩码应答等等。 IGMP 协议主要是用于支持主机和路由器进行多播。

(3) 传输层协议

传输层主要包括 UDP 协议和 TCP 协议。 用户数据报协议 UDP 向应用程序提供了一种面向数据报的传输层协议，每个 UDP 数据报被组装成一个 IP 数据分组，UDP 不提供可靠性，它不保证数据报能准确无误的到达目的地。 传输控

制协议 TCP 主要是专门设计用在不可靠的 Internet 网络层上提供面向连接的可靠的字节流通信服务，TCP 被设计成能动态满足互联网的要求并能处理各种错误的可靠性协议，该协议的使用范围很广，很多应用层的软件都是基于 TCP 协议。

（4）应用层协议

建立在 TCP/IP 的应用层协议主要有 Telnet 协议、文件传输协议 FTP、简单邮件协议 SMTP、简单的网络管理协议 SNMP 和超文本连接协议 HTTP 等等。这些应用都是建立在传输层之上的协议，用户可以根据应用的需要采纳不同的应用协议。

1.2 嵌入式系统接入 Internet

为了实现多个系统之间的信息交流，可以利用 CAN、RS-232、RS-485 等总线将各个系统连接，但这种网络的有效半径比较有限，通信的速度也不高，有关的通信协议也比较少，并且一般是孤立于 Internet 以外的。随着 Internet 网络软、硬件的快速发展，互联网用户正在以指数增长，Internet 现已成为社会重要的基础信息设施之一，是信息流通的重要渠道，如果嵌入式系统能够连接到 Internet 上面，则可以方便、低廉地将信息传送到几乎世界上的任何一个地方。以太网经过 20 多年的发展，已经成为当今互联网络中底层链接不可缺少的部分。TCP/IP 协议栈是一个非常复杂和庞大的系统，它是 Internet 互联网安全可靠通讯的重要组成部分，通常在通用计算机系统和操作系统下实现。而各种家电设备、PDA、仪器仪表、工业生产中数据的采集与控制等设备也正在逐渐地走向网络化，以便共

享互联网络中庞大的资源。在某些应用领域，嵌入式设备在价格、体积及实时性等方面，有着通用计算机无法比拟的优点，因此，嵌入式设备的网络化开发有着广阔的前景。高档的嵌入式处理器和嵌入式操作系统使得嵌入式系统方便的接入 Internet，但是它的成本也增加了。目前，8 位处理器仍然占据最主要的市场，特别是 51 系列。但是 51 系列单片机的资源太低，特别是 RAM 和 ROM 大小，所以必须外部扩展 RAM 和 ROM 存储器。当前，16 位单片机比较流行，特别是凌阳、TI 的 430 系列单片机，它们内部硬件资源丰富、功耗低、RAM 和 ROM 容量较大，因此在此种单片机上开发嵌入式 TCP/IP 协议无需额外扩展硬件即可满足要求，比较方便灵活。本系统采用性价比较高的 MSP430F149 单片机。因此以 430 系列处理器为基础，嵌入式系统直接接入 Internet，除了在网络硬件接口建设方面的开发以外，一个重要的问题就是根据嵌入式系统的资源有限，功能单一的特点开发一套微型 TCP/IP 互联网络协议。如何在价格低廉、资源有限的嵌入式系统的环境中实现网络通讯功能已经成为嵌入式网络开发人员面对的重要问题[22]。我们设计的微型 TCP/IP 协议栈能满足这一需要，能够完成常用的网络通讯功能。

一般在单片机接入 Internet 的系统里只实现与需要有关的部分，而不使用的协议则一概不支持。如文件共享 (SMB：Server Message Block) 协议，在 UNIX、WINDOWS 都支持，但单片机上却没有必要。一般只能在单片机中实现：ARP、IP、ICMP、TCP、UDP 这些协议，而更高层的协议，HTTP、SMTP、FTP 一般是不需要支持的。虽然有些单片机例如 AVR 上网方案实现了 HTTP、SMTP、FTP

协议，但实用性不太，因为 AVR 上网方案用的是 MEGA103，高昂的造价使得 AVR 上网方案没有得到广泛的应用。单片机应用 TCP/IP 协议大多是为了完成数据采集和数据传输，而不是网页浏览、文件传输这些功能。就对某一协议而言，如 ARP 协议，UNIX 系统支持以太网、令牌环等网络的 ARP，但单片机里只支持以太网，也就是说，对于某一协议，也有可能要作简化。IP 包最大可以为 65k，可以分段传输，而在单片机里根本无法容纳如此大的数据包，因此一般是不支持分段的。单片机一般采用发送小数据包的方式，以避免分段。

1.3 嵌入式 TCP/IP 协议栈 的特点

TCP/IP 协议栈已经成为开放系统互联的标准，与其他任何协议相比，TCP/IP 协议能够提供更好的交互操作性能，可以在大多数系统中使用，并且它可兼容多种网络技术，因而 TCP/IP 非常流行并且已经得到了广泛使用。微型 TCP/IP 协议栈具有 TCP/IP 协议栈最基本功能。运行于以太网环境下，底层可以处理 ARP 请求；IP 层支持两种协议：ICMP、TCP。其中 ICMP 协议能对 ECHO 命令响应，TCP 协议提供了一个快速的、可靠的连接。高层协议能够稳定的运行于 TCP 之上，比如：HTTP、SMTP、TELNET 等等。各层分别采用独立的模块，整个系统具有紧凑、轻便、模块化、高效和可升级等特点。由于微型 TCP/IP 协议栈主要应用于基于单片机的嵌入式系统，该系统的各种资源有限，如 CPU 的处理

速度、字长、RAM 和 ROM 存储器的容量、接口数量等和通用计算机相比有很大的差距。因此，如何使微型 TCP/IP 协议栈做到精细、通讯可靠、功能相对完善，而且又能发挥单片机的特点成为微型 TCP/IP 协议栈设计的关键问题。为了减少系统的资源开销，微型 TCP/IP 协议栈在功能上根据需要进行了精简。我们对微型 TCP/IP 协议栈做了如下限制：

(1) 每次仅仅进行一个 TCP 会话，UDP 本文没有进行设计。

(2) 不能重组接收的 IP 帧。

(3) 在错误的顺序号下不能缓冲 TCP 数据段。

(4) 不支持服务类型和安全选项。

(5) 忽略接收的 TCP 选项。

在满足系统通信要求的情况下，本文只实现了 ARP 协议、ICMP 协议、IP 协议、TCP 协议。

2 精简 TCP/IP 协议栈的实现

我们从两个方面简要讨论一下系统的实现：软件模型、帧结构和具体实现。

2.1 软件模型

实现 TCP/IP 协议栈软件，如图 2.1 所示。为了有效利用单片机系统有限的资源并提高效率，各层协议的具体实现和通常的 TCP/IP 协议相比作了较大的简化，但它具有最基本的功能。

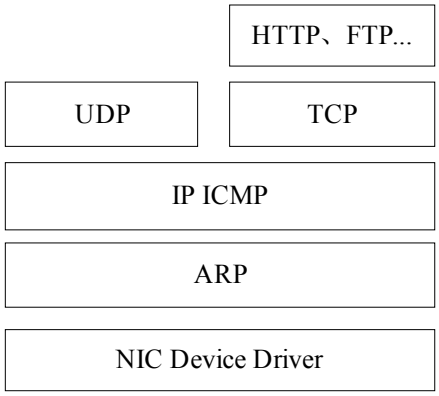

图 2.1 微型 TCP/IP 协议栈软件模型

Fig. 2.1 Software model of micro TCP/IP protocol

最底层是网络接口层，它主要定义了以太网控制器 CS8900A 的工作方式，以太网帧的封装和分析，发送以及接收等，我们使用了 Cirrus Logic 公司的以太网控制器芯片 CS8900A。它完成了物理帧的封装和解封装。第二层是 Internet 网络层，它主要提供了不可靠的、尽最大努力的传输，完成了 IP 数据报的封装，IP 头的分析，并根据帧的类型进行相应的处理，ICMP、IGMP 等。第三层是传输层，主要是 TCP，它对 Internet 上提供点到点的、可靠的服务，在这层上只允许单个连接。它实现了 RFC793 中相应的功能，包括三次握手建立连接，各种状态

之间的转换，连接的撤销、数据的传输等等。最上层是应用层，解决用户特定的应用，而不用关心底层的具体实现[25]。在我们的系统中应用 HTTP（Hypertext Transfer Protocol：超文本传输协议）协议实现了一个简单的 web 服务，应用 ESMTP（Enhanced Simple Message Transfer Protocol：增强简单邮件传输协议）实现了一个简单的邮件传输接收功能。在电表数据传输上，采用的是本文为该系统设计的协议，这个协议类似 SMTP。

2.2 各协 议的帧 结构

（1）ARP 帧结构

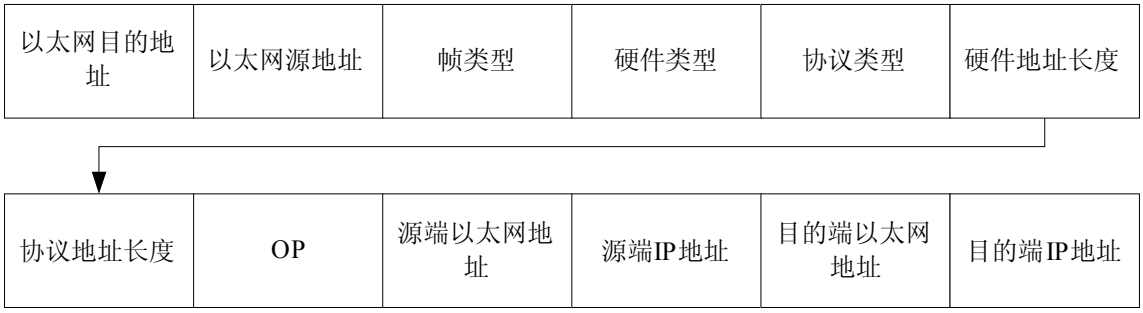

图 2.2 ARP 请求/应答数据帧结构

Fig. 2.2 ARP request/answer data structure

以太网报头的前两个字段是以太网目的地址和以太网源地址。目的地址全 1 的特殊地址是广播地址。ARP 请求或应答数据帧如图 2.2 所示。对 ARP 请求和应答来说，帧类型字段的值为 0x0806。硬件类型字段的值表示硬件地址的类型，

为 1 表示以太网地址。协议类型字段的值表示映射的协议地址类型，它的值为 0x8000 表示 IP 地址。硬件地址长度和协议地址长度指出硬件地址和协议地址的长度，以字节为单位。对以太网上的 IP 地址的 ARP 请求和应答来说，它们的值为 6 和 4。操作字段指出四种操作类型，1 表示 ARP 请求、2 表示 ARP 应答、3 表示 RARP 请求、4 表示 RARP 应答。

(2) IP 帧结构

表 2.1 IP 帧结构

Tab. 2.1 Structure of IP frame

<table>
<tr><td>4 位版本</td><td>4 位首部长度</td><td>8 位服务类型
(TOS)</td><td colspan="2">16 位总长度</td></tr>
<tr><td colspan="3">16 位标识</td><td>3 位标志</td><td>13 位片偏移</td></tr>
<tr><td colspan="2">8 位生存时间</td><td>8 位协议</td><td colspan="2">16 位首部校验和</td></tr>
<tr><td colspan="5">32 位源 IP 地址</td></tr>
<tr><td colspan="5">32 位目的 IP 地址</td></tr>
<tr><td colspan="5">选项（如果有）</td></tr>
<tr><td colspan="5">数据</td></tr>
</table>

如表 2.1 所示，其中协议版本号位 4。首部长度指的是首部占 32 位字的数目，普通的 IP 数据报字段的值是 5。服务类型 TOS 字段包括一个 3 位的优先权子字段(现在已被忽略)，4 位的 TOS 子字段和 1 位未用(必须置 0)，其中 4 位的 TOS 分别代表：最小延迟、最大吞吐量、最高可靠性和最小费用。4 位中只能置其中 1 位。如果所有 4 位均为 0，那么就意味着一般服务；总长度字段是指整个 IP 数据

报的长度，以字节为单位；标志字段用于唯一的标识主机发送的每一个 IP 数据报。3 位标志中，首位保留位(必须置 0)，接着第 2 位用于标志本 IP 数据报不进行分片，第 3 位用于表示还有进一步的的分组。接着 13 位片偏移是指分组在当前数据报的位置。TTL (Time To Live) 生存时间字段设置了数据报可以经过的最多路由器数，通常是 32 或 64；首部校验和字段是根据 IP 首部计算的校验和码，采用的算法是行二进制反码求和。最后一个字段是任意项，是数据报中的一个可变长的可选信息。

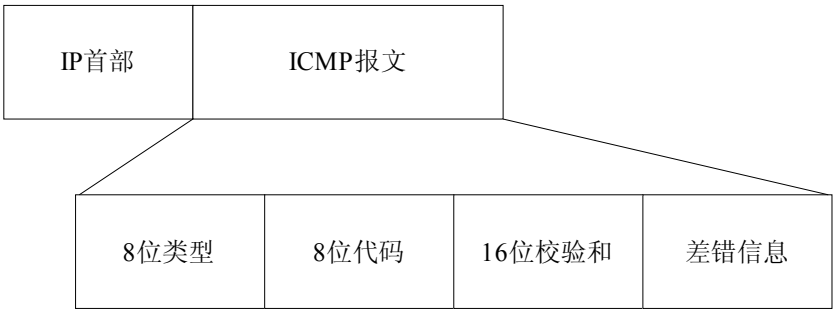

图 2.3 ICMP 数据帧格式

Fig. 2.3 Format of ICMP data frame

(3) ICMP 帧格式

如图 2.3 所示，ICMP 可用于差错控制和信息查询，在嵌入式设计系统中，主要实现差错的控制，而其他部分可以省略。ICMP 差错信息由 8 位类型和 8 位代码共同组成。如果 ICMP 函数处理时，将这些不相关的的信息都标志为其他错误信息，可以做简单的记录也可以丢弃。

(4) TCP 帧格式

TCP 提供一种面向连接的、可靠的字节流传送服务。TCP 数据包的结构如表 2.2 所示。

表 2.2 TCP 数据包结构

Tab. 2.2 Structure of TCP data package

<table>
<tr><td colspan="3">源端口</td><td>目的端口</td></tr>
<tr><td colspan="4">序号</td></tr>
<tr><td colspan="4">确认区号</td></tr>
<tr><td>首部长度</td><td>保留</td><td>标志</td><td>窗口</td></tr>
<tr><td colspan="3">校验和</td><td>紧急指针</td></tr>
<tr><td colspan="4">可选项</td></tr>
<tr><td colspan="4">数据</td></tr>
</table>

每个 TCP 段都包含源端和目的端的端口号，用于寻找发端和收端的应用进程。这两个值加上 IP 首部中的源端 IP 地址和目的端 IP 地址唯一确定一个 TCP 连接。序号用来标识从 TCP 发端向 TCP 收端发送的数据字节流。它表示在这个报文段中的第一个数据字节。确认序号包含发送确认的一端所期望收到的下一个序号。首部长度给出首部中 32 位字的数目。标志是 6 位数据，它表示了当前 TCP 数据帧的状态，TCP 协议层根据这个状态进行相应的处理。紧急指针是一个正的偏移量，与序号字段中的值相加表示紧急数据最后一个自己的序号。

2.3 网卡驱动及精简 TCP/IP 协议栈的实现

(1) 网卡驱动

由于MSP430系列单片机总线不能扩展，因此需要用IO管脚来模拟出地址总线和数据总线。

CS8900A的配置：复位采用软复位；中断采用查询方式；数据总线为SD0-SD7；地址总线是SA0-SA3。

接入以太网首要工作就是编写网卡芯片的驱动程序，给单片机提供读写CS8900A寄存器的接口、产生读写CS8900A的时钟等的函数。以太网程序包含了网络接口的各种配置，包括MAC(介质访问控制)地址的配置。首先要初始化MSP430单片机的读写引脚，开始配置网卡芯片的寄存器状态，然后复位，等待复位成功之后，MSP430F149把本机的MAC地址以配置序列字的形式发CS8900A，CS8900A把MAC地址存储在寄存器中，即完成了网卡芯片的初始化工作。单片机就可以读写CS8900A了。

网卡驱动部分实现的主要三个函数就是：CS8900A初始化函数、CS8900A读数据函数和CS8900A写数据函数。

TCP/IP协议实现是整个网络终端软件的关键，定义了使用网络来收发数据的协议，使用以太网模块所提供的各种读写函数来接收或发送数据，并给应用层提供简单易用的API接口函数。

TCP/IP协议是个非常复杂的协议，简略的讲可以分为不同层次进行开发，每一层负责不同的通信功能。

系统的工作过程如图所示。当系统加电时，系统首先读取保存在

MSP430F149 内部 Flash 的 MAC 地址、IP 地址、子网掩码和默认网关等参数。然后对 CS8900A 以太网控制器进行初始化，初始化完成后，系统进入监听状态，被动的等待连接的到来。当 CS8900A 接收到请求数据包后，系统主机开始接收数据包。根据以太帧（即刚刚接收的数据包）中包含的帧的类型分别进行 ARP、ICMP、TCP 的处理，并准备相应的回答帧，交给以太网控制器，由以太网控制器发送至以太网络。处理结束后，系统继续等待接收下一个数据包，如此循环。

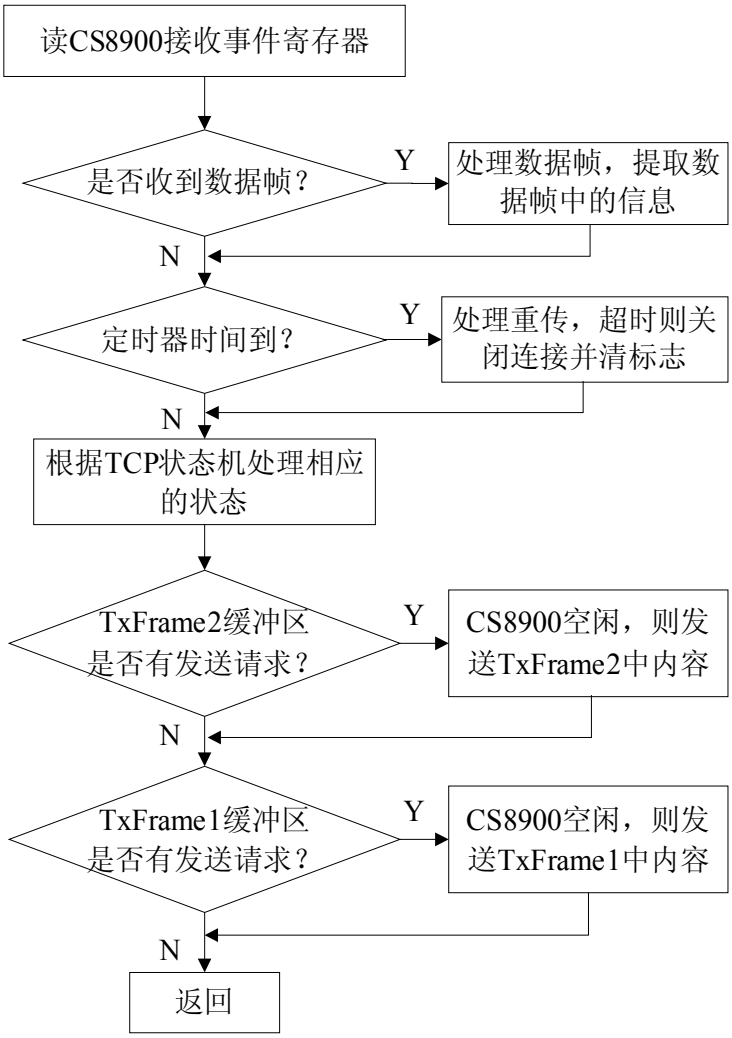

图2.4 DoNetworkStuff()函数流程图

Fig. 2.4 Flow chart of DoNetworkStuff() function

TCP/IP 模块实现中，最重要的函数是 DoNetworkStuff()，该函数的主要作用是进行 TCP 事件调度处理，需要周期性地被 main()函数调用。在模块中定义了枚举变量 TCPStateMachine 作为 TCP 状态机（参照 RFC793），而函数 DoNetworkStuff()则根据 TCP 状态机状态来进行相应的处理工作。该函数工作流程如图 2.4 所示。DoNetworkStuff()函数被调用得周期越短，处理速度就越快，响应速度也越快。

为了高效的处理网络通信中的数据，需要为 TCP/IP 模块在 SDRAM 中分配两个数据缓冲区 TxTCPBuffer 和 RxTCPBuffer。其中，TxTCPBuffer 的主要作用是给要发送的 TCP 数据帧提供缓冲内存。另外还分配了两个数据缓冲区 TxFrame1 和 TxFrame2，TxFrame1 为以太网、IP、TCP 帧头提供缓冲内存，而 TxFrame2 的主要作用是给 ARP、ICMP 协议帧提供缓冲内存，发送的数据帧连接在 TxFrame1 之后。当主程序需要发送数据时，则将数据拷贝到 TxTCPBuffer 中，并置位发送 TxFrame1 的标志，DoNetworkStuff()函数根据标志处理发送整个 TxFrame1 缓冲区中的数据。RxTCPBuffer 给接收到的 TCP 数据提供缓冲内存，所分配的缓冲区越大，收发速度就越快，因为这样可以减少因缓冲区不足造成的处理时间延迟。根据实际应用分配缓冲区的大小，本应用由于仅是对话应答，数据量很小，所以将收发缓冲区大小都设定为 256 字节。

可以通过调用封装好的 API 接口函数 SocketClient()或 SocketServer()建立网络通信连接。函数 SocketServer()的作用主要是用于监听是否有连接接入并把数据包送入接收数据缓冲区。函数 SocketClient()的主要作用是，根据本机 IP 地址、MAC 地址、端口号、远程主机 IP 地址和端口号，建立一个 Socket 连接，建立完成之后，把要发送的数据送入发送缓冲区并调用发送函数，就可以将发送数据发送到远程主机了。可以通过变量 SocketStatus 了解当前连接的状态。当数据发送完毕，通过函数 SocketClose()关闭连接。

在 TCP 传输控制协议中，两个主机建立连接状态后，需要交换初始序号 ISN。在数据包收发的过程当中，为了保证通信可靠，防止丢包和延时就需要定时来进行 TCP 重传与超时处理，这个定时时间由 MSP430F149 的定时器 Timer_A 来提供，同时 Timer_A 还产生了初始序号。在 TCPInit() 函数中进行定时器的初始化。这些处理都是为了保证 TCP 传输的可靠性。

在 TCP/IP 协议的实现过程中，将用户编程接口统一编写成了 API 函数，将精简协议栈提供的功能封装在 API 函数中。而对于 TCP，创建了比较熟悉的 Socket 结构。

2.4 重要的宏定义及数据结构

```
// IP 地址、子网掩码、网关宏定义

#define MYIP_1          192          // IP 地址
```

```
#define MYIP_2          168
#define MYIP_3          0
#define MYIP_4          1

#define SUBMASK_1       255             // 子网掩码
#define SUBMASK_2       255
#define SUBMASK_3       255
#define SUBMASK_4       0

#define GWIP_1          192             // 网关 IP
#define GWIP_2          168
#define GWIP_3          0
#define GWIP_4          250
// 根据 RFC793 定义的 TCP 状态机
typedef enum {
  CLOSED,
  LISTENING,
  SYN_SENT,
  SYN_RECD,
  ESTABLISHED,
  FIN_WAIT_1,
```

```
  FIN_WAIT_2,
  CLOSE_WAIT,
  CLOSING,
  LAST_ACK,
  TIME_WAIT
} TTCPStateMachine;
//记录上一个数据帧类型用于重传机制
typedef enum {
  ARP_REQUEST,
  TCP_SYN_FRAME,
  TCP_SYN_ACK_FRAME,
  TCP_FIN_FRAME,
  TCP_DATA_FRAME
} TLastFrameSent;


//定义了最大的接收与发送数据包的字节长度

#define MAX_TCP_TX_DATA_SIZE 128  //最大的发送 TCP 字节数

#define MAX_TCP_RX_DATA_SIZE 128  // 最大的接收 TCP 字节数
```

2.5 网络终端应用层软件设计

应用中，只需调用 API 中的 Socket 函数来实现网络部分的功能。

3 用户终端网络模块硬件设计

3.1 用户终端网络模块结构

网络模块的结构如图 3.1 所示。

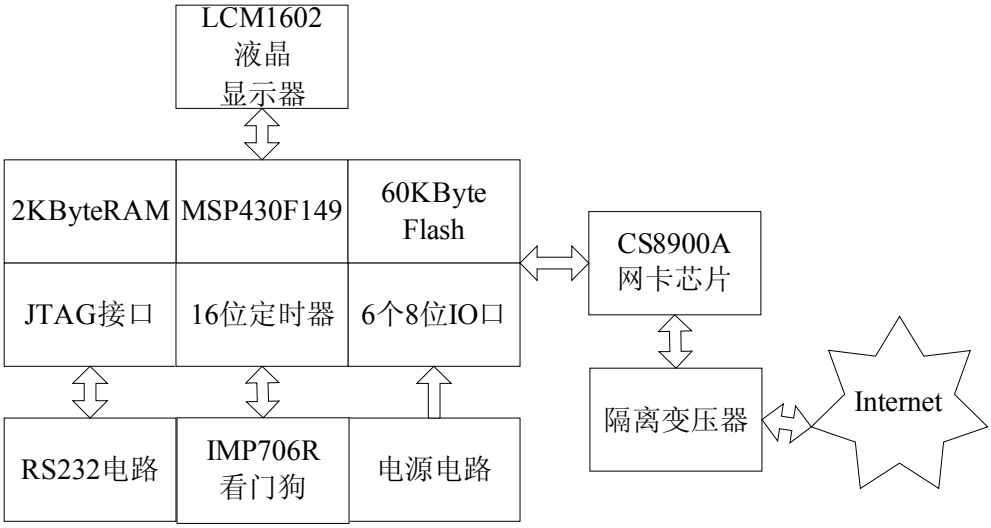

图 3.1 用户网络终端结构图

Fig. 2.8 Structure of user’s network terminal

MSP430F149 单片机是 TI 公司生产的高性能超低功耗的 16 位单片机。内部

集成了 60KByte 的 Flash 存储器、2KByte 的 RAM、2 个 16 位定时器、一个看门狗、12 位 AD 等等。它是本系统用户终端网络模块的核心元器件。因为它丰富而强大的内部资源使得用户终端无论从结构上还是性能或者功耗上都非常理想。在线仿真及程序下载采用 JTAG 方式。

CS8900A 是 Cyrital 公司生产的高度集成的网卡芯片，是单芯片网络的解决方案。它作为本系统用户终端的主要元器件，用来实现 TCP/IP 协议所需要的所有的数字电路及模拟电路，即实现了协议的物理层。

LCM1602 是当前比较流行的的液晶显示模块。他能够显示 16×2 个字符，与单片机通信可以采用 4 数据线或 8 数据线，可根据单片机端口数量而选择。具有背光功能，显示控制方式多样且灵活，并且价格低廉。它用来显示当前的用电量信息以及其他的一些需要用户了解的信息。

用户网络终端的详细电路图见附录 A。

3.2 MSP430 系列单片机

(1) MSP430 系列单片机是由美国德州仪器 (TI) 开发研制的一种具有超低功耗特点的功能强大的 16 位 RISC 结构单片机。它有 MSP430x1x、MSP430x2x、MSP430x3x、MSP430x4x 等几大系列；其中 MSP430Fx 系列具有 FLASH 存储器，使系统设计、开发调试和实际应用都变得比较方便。MSP430F1x 系列是 MSP430FLASH 型单片机中的一种，其具有以下特点：

① 超低功耗。运行在 1MHz 时钟条件下时，工作电流视工作模式不同为 0.1～400，工作电压为 1.8～3.6V。

② 强大的处理能力。MSP430F1x 具有丰富的寻址方式(7 种元操作数，4 种目的操作数)，但只需要简洁的 27 条指令；片内寄存器数量多，存储器可实现多种运算；具有高效的查表处理方法。这些特点保证了可以编制出高效的程序。MSP430F1x 具有较多的中断源，并且可以任意嵌套，使用时方便灵活。当系统处于省电模式时，只需要 6ms 就能用中断请求将它唤醒。

③ 丰富的片上外围模块。MSP430F1x 中的各成员集成了较多的片上外围资源，如：12 位 ADC、精密模拟比较器、硬件乘法器、带有捕获/比较寄存器的 16 位定时器、看门狗、可实现异步/同步及多址访问的串行通信接口、数十个可实现方向设置及中断功能的并行输入输出端口等。

④ 方便高效的开发方式。具有 FLASH 存储器的 MSP430F1x 使它的开发工具相当简便。利用单片机本身具有的 JTAG 接口或片内 BOOT ROM，可以在一台 PC 机和一个结构小巧的 JTAG 控制器的帮助下实现程序的下载，完成程序调试。

⑤ MSP430F1x 的各个型号大多有性能相同而存储器不同的 ROM 型、OTP 型，以适应产品在设计、开发、生产的各个不同阶段的需要。

⑥ 适应工业级运行环境 MSP430F1x 的运行环境温度范围为-40～+85℃，所设计的产品适合运行于工业环境下。这个系列目前有 MSP430F11x、MSP430F11x1、MSP430F13x、MSP430F14x、MSP430F15x、MSP430

F16x 等型号，所集成的存储器容量及外围模块各不相同，可适应不同需要。

(2) MSP430 系列单片机的内部结构

MSP430 系列单片机采用存储器-存储器结构，即用一个公共的空间对全部功能模块寻址，同时用精简指令组对全部功能模块进行操作。由于不同型号的单片机具有不同的外围模块和不同大小的存储器，因此在内部结构上有细微的差别。图 3.2 是 MSP430F14x 系列单片机内部结构图。

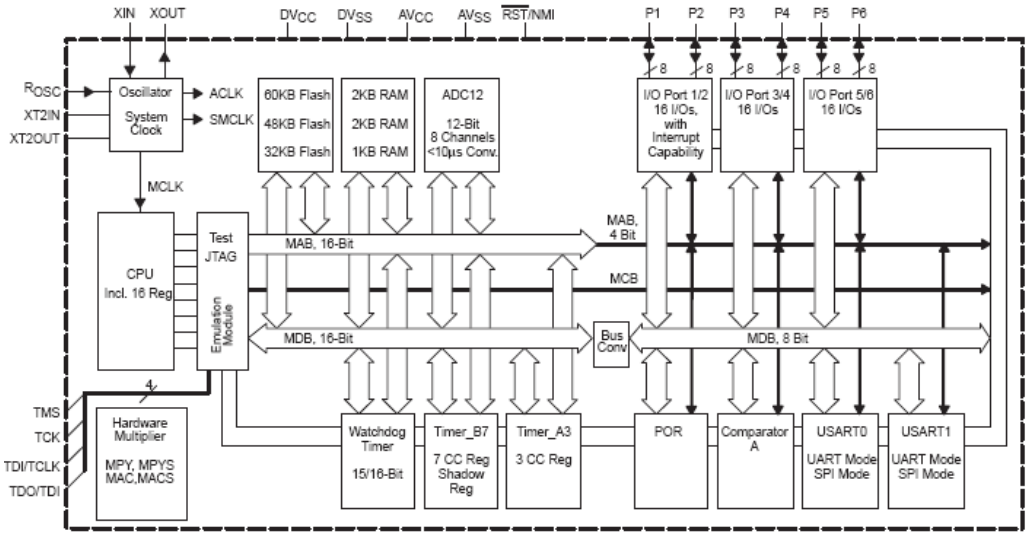

图 3.2 MSP430F14x 系列单片机内部结构图

Fig. 3.2 Internal structure of MSP430F14x

(3) MSP430 系列单片机的 CPU

MSP430 系列单片机的 CPU 采用正交的精简指令集，由 16 位 ALU、指令控制逻辑和 16 个寄存器组成，如图 3.3 所示。

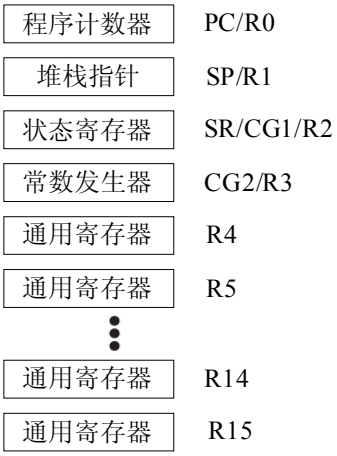

图 3.3 MSP430 的 CPU 结构

Fig. 3.3 CPU structure of MSP430

寄存器中有 4 个具有特殊用途，即：程序计数器 PC/R0、堆栈指针 SP/R1、状态寄存器和常数发生器 SR/CG1/R2、常数发生器 CG2/R3。除常数发生器 CG1 和 CG2，所有寄存器都可作为通用寄存器，用所有指令操作。

(4) MSP430 系列单片机的存储空间

MSP430 单片机的存储空间采用“冯·诺依曼”结构，ROM、RAM 和外围模块由同一组地址及数据总线连接在同一地址空间中，因此可以用相同的指令访问 ROM、RAM 和外围模块，也可以执行 RAM 中的程序代码。无论 ROM、RAM、SFR 或外围模块，都位于同一公共地址空间中，寻址空间为 64KB。图 3.4 为 MSP430 存储空间结构图。

数据总线可以是 16 位或 8 位宽度。16 位外围模块用 16 位数据宽度(字)访问，8 位外围模块用 8 位数据宽度(字节)访问。程序存储器(ROM)和数据存储器

(RAM) 可以用字也可以用字节进行访问。

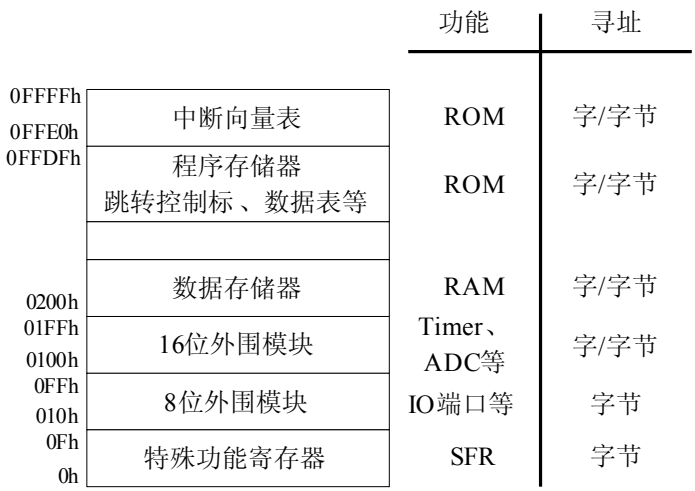

图 3.4 MSP430 存储空间基本结构

Fig. 3.4 Memory structure of MSP430

字节数据可以定位在偶地址或奇地址。字数据定位在偶地址，低字节在偶地址，高字节在奇地址，因此，字指令访问时总是用偶地址。

ROM 的开始地址因 ROM 体积大小的不同而不同。中断向量位于 ROM 的最高 16 字地址区。其中最高优先级的向量位于最高的 ROM 字地址 0FFFEh。如图 3.4 所示。

(5) MSP430 系列单片机的中断系统

MSP430 系列单片机有丰富的中断资源，这些中断可以分为四类：

① 系统复位

② 可屏蔽中断(maskable)

表 3.1 F13x 及 F14x 中断向量表

Tab. 3.1 Interrupt vector of F13x and F14x

中断源	中断标志	系统中断	地址	优先级
上电、外部复位				
看门狗复位	WDTIFG	复位	0FFFE	15,最高
FLASH	KEYV			
NMI	NMIIFG	(非)屏蔽		
振荡器故障	OFIFG	(非)屏蔽	0FFFC	14
FLASH 存储器非法访问	ACCVIFG	(非)屏蔽		
Timer_B7	BCCIFG0	可屏蔽	0FFFA	13
Timer_B7	BCCIFG1	可屏蔽	0FFF8	12
	TBIFG			
比较器 A	CMPAIFG	可屏蔽	0FFF6	11
WDT	WDTIFG	可屏蔽	0FFF4	10
USART0 接收	URXIFG0	可屏蔽	0FFF2	9
USART0 发送	UTXIFG0	可屏蔽	0FFF0	8
ADC	ADCIFG	可屏蔽	0FFEE	7
Timer_A3	CCIFG0	可屏蔽	0FFEC	6
	CCIFG1			
Timer_A3	CCIFG2	可屏蔽	0FFEA	5
	TAIFG			
P1	P1IFG.0-7	可屏蔽	0FFE8	4
USART1 接收	URXIFG1	可屏蔽	0FFE6	3
USART1 发送	UTXIFG1	可屏蔽	0FFE4	2
P2	P2IFG.0-7	可屏蔽	0FFE2	1
		可屏蔽	0FFE0	0，最低

③ 非屏蔽中断(non-maskable)

④ (非)屏蔽中断((non)-maskable)

系统复位包括：上电复位(POR)和上电清除(PUC)。上电复位在：芯片上电和在RST/NMI引脚上出现低电平信号(RST/NMI被设置成复位模式)时产生。上电清除在：上电复位、看门狗计时溢出、看门狗定时器写入错误的安全键值和FLASH存储器写入错误的安全键值时产生。上电复位总是会产生上电清除，但上电清除不会产生上电复位。

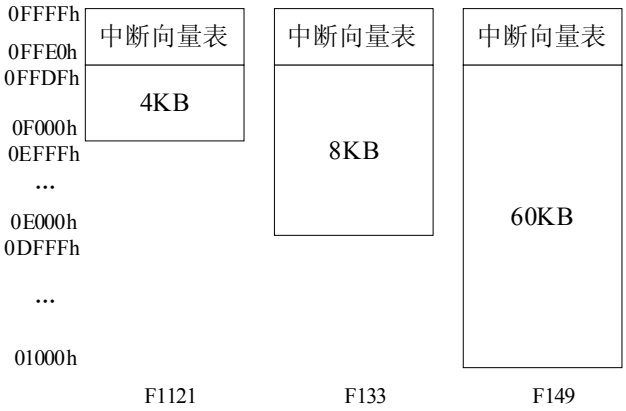

图3.5不同型号ROM地址分布

Fig. 3.5 ROM distributing of different chip

非可屏蔽中断是无法屏蔽的，这些中断既无中断允许位，也不会受通用中断允许位(GIE)的影响。

(非)屏蔽中断不能用通用中断允许位(GIE)屏蔽，但是可以用各自的中断允许位控制。当一个(非)屏蔽中断请求被接受时，相应的中断允许位就自动复位，因此也禁止了新的中断请求。RETI指令不影响相应的(非)屏蔽中断的允许位，因此必须用软件在中断服务程序中的RETI指令前将中断允许位置位，以保证在中断服务结束后能再次被允许。(非)屏蔽中断可以有以下条件产生：

① 当选择 NMI 模式时，RST/NMI 引脚产生跳变沿。

② 当振荡器失效中断允许时，发生时钟失效。

③ 当非法访问中断允许时，发生对 FLASH 存储器的非法访问。

中断向量表位于 ROM 的最高地址的 16 个字。每一个中断源有唯一的中断向量与之对应，但一个中断向量所对应的中断源并不唯一。中断向量在中断向量表中的地址越大对应中断的优先级越高。由于开发时间的先后不同，不同型号的单片机的中断向量表设计有较大的差别，表 2.6 是 MSP430F13x 及 MSP430F14x 的中断向量表。

(6) 工作模式

MSP430 是为超低功耗应用开发的，它有多种工作模式来实现这一功能。MSP430 的各种工作模式状态以强有力的方式支持超低功耗的各种要求，这是通过各模块的智能化运行管理和 CPU 的状态组合而得到的。一个中断事件可以将系统从各种工作模式中唤醒，而 RETI 指令又使 MSP430 返回到中断事件发生前的工作模式[16]。

MSP430 单片机通过控制位于状态寄存器(SR)中的 4 个功能位：SCG1(系统时钟发生器控制位 1)、SCG0(系统时钟发生器控制位 0)、OSCOff(晶振关闭位)和 CPUOff(CPU 关闭位)来控制单片机工作在不同的节能模式。如表 3.2 所示，通过软件可组合成 6 种工作模式：

① 活动模式(AM)：SCG1=0，SCG0=0，OSCOff=0，CPUOff=0；CPU、时钟处于活动状态。

② 低功耗模式 0(LPM0)：SCG1=0，SCG0=0，OSCOff=0，CPUOff=1；CPU 被禁止；系统主时钟(MCLK)被禁止；系统从时钟(SMCLK)、辅助时钟(ACLK)保持活动。

③ 低功耗模式 1(LPM1)：SCG1=0，SCG0=1，OSCOff=0，CPUOff=1；CPU 被禁止； MCLK 被禁止；如果数控振荡源(DCO)未作用于 MCLK 或 SMCLK，又处于活

动状态，则直流发生器(DC)被禁止，否则仍处于活动状态；SMCLK、ACLK 保持活动。

表 3.2 低功耗工作模式控制位

Tab. 3.2 The work mode control bit of low power consume

省电模式	SCG1	SCG0	OSCOff	CPUOff
LPM0	0	0	0	1
LPM1	0	1	0	1
LPM2	1	0	0	1
LPM3	1	1	0	1
LPM4	X	X	1	1

④ 低功耗模式 2(LPM2)：SCG1=1，SCG0=1，OSCOff=0，CPUOff=1；CPU 被禁止； MCLK 被禁止；SMCLK 被禁止；DCO 被禁止；ACLK 保持活动。

⑤ 低功耗模式 3(LPM3)：SCG1=1，SCG0=1，OSCOff=0，CPUOff=1；CPU 被禁止； MCLK 被禁止；SMCLK 被禁止；DCO 被禁止；DCO 的 DC 发生器被禁止；ACLK 保持活动。

⑥ 低功耗模式 4(LPM4)：SCG1=X，SCG0=X，OSCOff=1，CPUOff=1；CPU 被禁止； MCLK 被禁止；SMCLK 被禁止；DCO 被禁止；DCO 的 DC 发生器被禁止；ACLK 被禁止；晶振停止工作。

(7) JTAG 接口

MSP430 的 JTAG 接口是标准的 JTAG 仿真接口，仿真器通过 JTAG 接口可以调试系统的硬件和软件。JTAG 接口的连线图如图 3.6 所示。

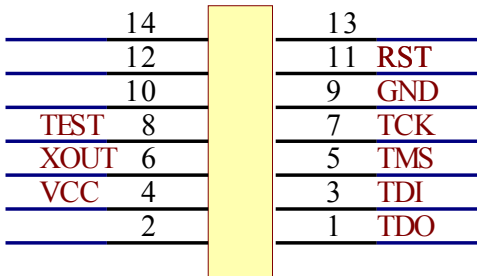

图 3.6 MSP430 的 JTAG 仿真接口

Fig. 3.6 The JTAG port for MSP430

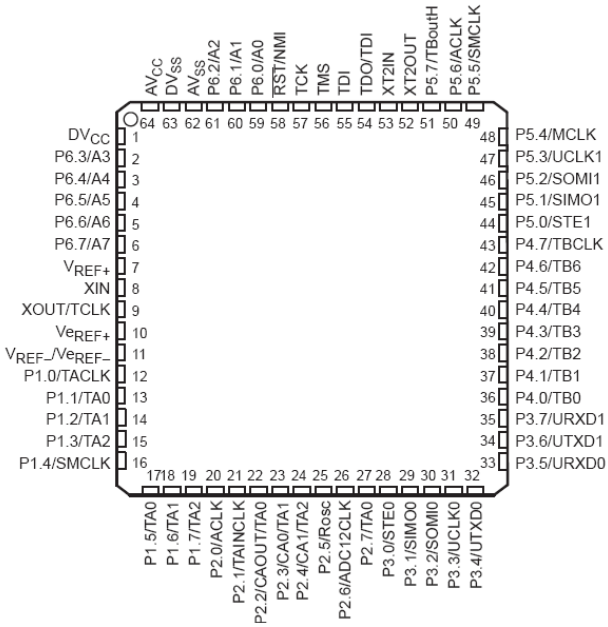

图 3.7 MSP430F149 引脚分布图

Fig. 3.7 MSP430F149 pinout

3.3 MSP430F149

MSP430F149 是 MSP430 FLASH 型单片机中资源较为丰富的一种，其有以下几部分组成：

（1）基础时钟模块，包括 1 个数控振荡器(DCO)和 2 个晶体振荡器。

（2）看门狗定时器 Watchdog Timer，可用作通用定时器。

（3）带有 3 个捕获/比较寄存器的 16 位定时器 Timer_A3。

（4）带有 7 个捕获/比较寄存器的 16 位定时器 Timer_B7。

（5）2 个具有中断功能的 8 位并行端口：P1 与 P2。

（6）4 个 8 位并行端口：P3、P4、P5 与 P6。

（7）模拟比较器 Comparator_A。

（8）12 位 A/D 转换器 ADC12。

（9）2 个串行通信接口：USART0 与 USART1。

（10）1 个硬件乘法器。

（11）60KB+256 字节 FLASH。

（12）2KB RAM。

MSP430F149 采用 64PM 封装形式，其引脚分布如图 3.7 所示。

3.4 网卡接口硬件设计

（1）网卡接口设计

基于以太网的嵌入式电表系统对网络功能的需求如下：

① 要求能够实现 TCP/IP 协议中物理链路层的全部功能。

② 具有较低的功耗，适合低功耗应用场合。

③ 性能稳定可靠。

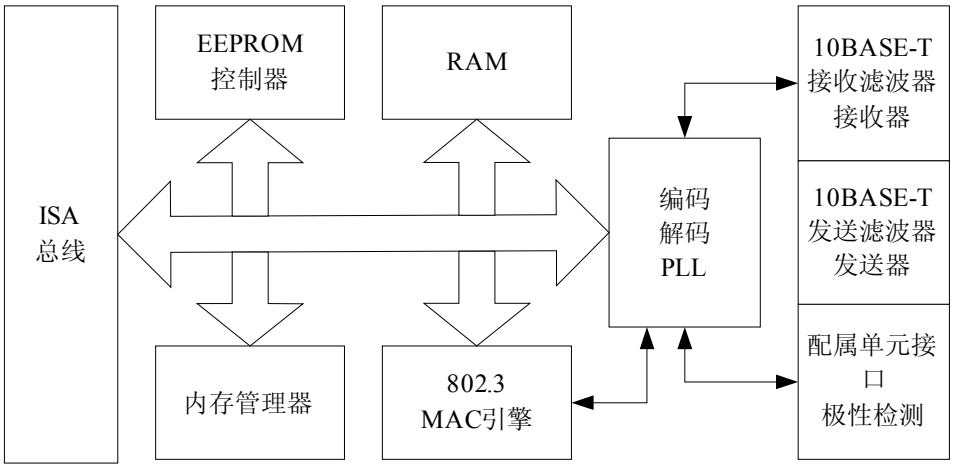

图 3.8 CS8900A 芯片内部结构框图

Fig 3.8 Inside structure of CS8900A

综合比较几个网卡芯片最终选择了 CS8900A 作为本系统的应用解决方案。如图 3.8 所示。

CS8900 以太网控制器是非常完整的以太网控制器，CS8900 芯片内部包含 RAM，包含 10BASE-T 传输与接收滤波器，和一个直接的 24mA 驱动的 ISA 总线接口。它还包括丰富的功能和配置选项。它独一无二的信息包页体系结构能够自动的适应网络传输模式的变化和可利用的系统资源，使得应用该芯片的系统

效率大为提高并且减小了 CPU 驱动它需要的资源。

(2) 网卡芯片 CS8900A

CS8900A 是一个完整的 16 位以太网控制器。它的设计目标是通过减小每帧数据在 ISA 总线和 CPU 处理上的时间来最优化系统的性能。CS8900A 是一个真正的单芯片、全双工通信、以太网解决方案。它包含了以太网电路所需要的所有的数字和模拟电路。主要的功能模块包括：一个直接的 ISA 总线接口；一个 802.3MAC 引擎；完整的超高速缓冲存储器 (RAM)；一个串行的 EEPROM 接口；一个完整的模拟前端以 10BASE-T 和 AUI 接口。常用的是 10BASE-T 接口。

① 直接的 ISA 总线接口

在 CS8900 内部包含的 ISA 总线接口拥有完整的 24mA 的驱动能力。它的配置选项包含一个可选择的四个中断和三个 DMA 通道 (在初始化的时候选择)。在存储模式里，它支持标准的和预备的总线周期不需要引入额外的等待状态。

② 完整的存储器

CS8900 芯片内部集成了一个 4K 字节页的芯片内部存储器。除掉了板上扩展额外存储器的芯片花费。跟大多数的网络控制器不同的是，在芯片内部，CS8900 的缓冲包含了整个的传输和接收数据帧，从而省去了原有的网络解决方案所必须的复杂的和低效率的内存管理结构。额外的，CS8900A 可以以存储模式、I/O 模式、或者 DMA 控制器来操作，极大的提高了系统的设计弹性。

③ 802.3 以太网 MAC 引擎

CS8900A 的 MAC 引擎与 IEEE802.3 以太网标准完全兼容，支持全双工操作。它能处理以太网传输和接收所有方面的情况。包括：冲突检测、序列产生和检测、CRC 序列产生和检测。可编程的 MAC 功能包括在冲突时刻的自动重发机制和自动的传输帧填充功能。

④ EEPROM 接口

CS8900 提供了一个简单的和高效的串行的 EEPROM 接口，它允许配置信息存储在一个可选择的 EEPROM 中，并且在上电时刻能够自动载入。这除去了所必须的昂贵的和笨重的开关和跳线。

⑤ 完整的模拟前端

CS8900 的模拟前端集成了一个曼切斯特编码和解码器、时钟恢复电路、10BASE-T 收发器、和一个完整的配属单元接口(AUI)。它提供了一个手动的或者自动的 10BASE-T 或 AUI 选择，并且提供了一个芯片内部驱动的三个 LED 接口分别指示连接状态、总线状态和以太网线路的活跃性。

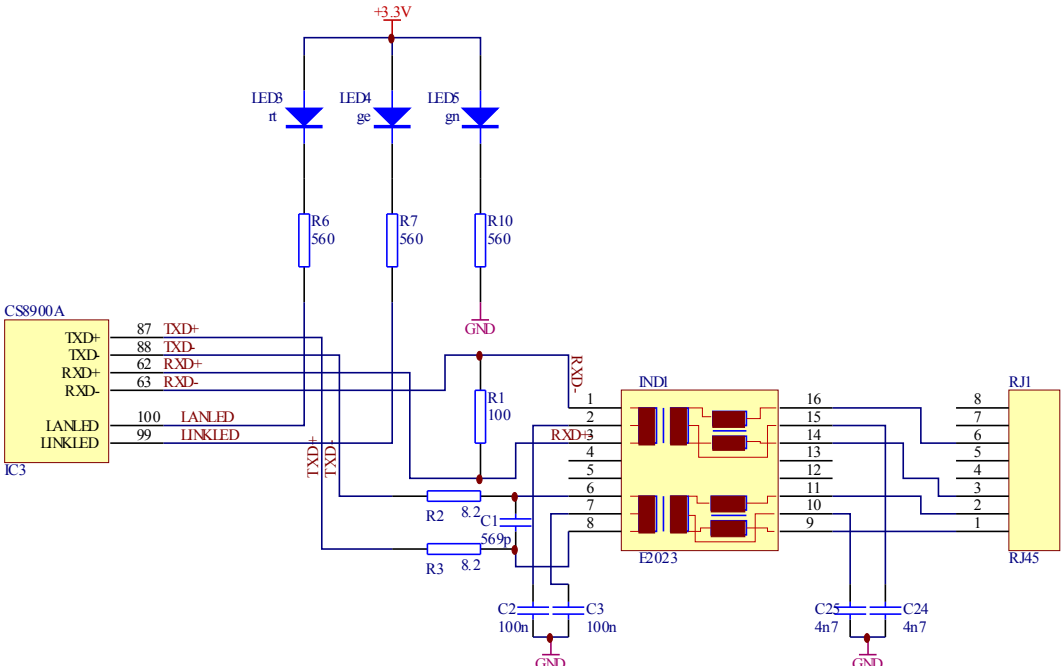

图 3.9 以太网的连接线路图

Fig3.9 Chart of Ethernet connection

10BASE-T 收发器包含了驱动器、接收器和模拟滤波器，允许直接连接价格便宜的隔离变压器。它支持 100、120 和 150 欧姆的屏蔽或非屏蔽电缆，延长了电缆长度，并且能够自动对极性颠倒进行检测和修正并能正确接收。以太网的的连接线路图如图 3.9 所示。E2030 为贴片封装的隔离变压器。

3.5 液晶 显示模 块 LCM1602

(1) LCM1602 简介

液晶显示电路采用的是比较流行的 LCM1602 液晶显示模块。LCM1602 为兼容的液晶显示模块，支持 5×7 点阵和 2 行×16 字符两种模式，背光亮度和显示

对比度可调，是一种功能较简易、价格较便宜的液晶显示器件。它由液晶显示器和驱动器两部分组成，单片机通过写控制字方式访问驱动器来实现对显示屏的控制。可选择的带有背光，驱动电压为 4.5 到 5.5V，无背光驱动电流为 2mA。

其与单片机的接口如图 3.10 所示。

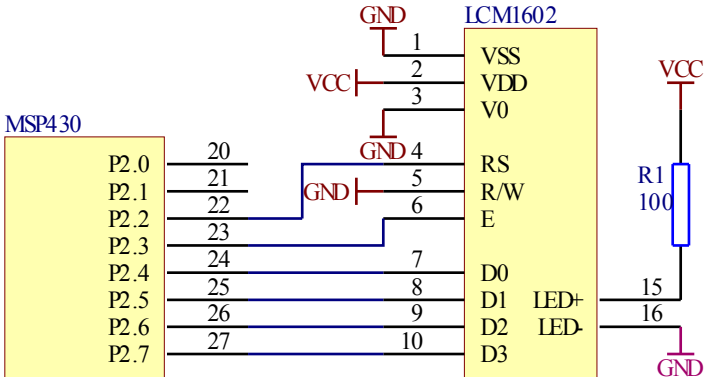

图 3.10 LCM1602 与 MSP430 单片机的接口示意图

Fig 3.10 Interface Chart of LCM1602 and MSP430 MCU

(2) LCM1602 的引脚功能表

如表 3.3 所示。

表 3.3 LCM1602 引脚说明

Tab. 3.3 Pin description of LCM1602

编号	标示	引脚说明
1	VSS	电源地
2	VDD	电源正极
3	V0	液晶显示偏压
4	RS	指令/数据选择端(H/L)
5	R/W	读/写选择端(H/L)
6	E	使能信号
7	D0～D7	8 位数据总线
8	LED+	背光源正极
9	LED-	背光源负极

(3) LCM1602 的基本操作时序

① 写指令：E＝高脉冲，RS＝L，RW＝L，D0～D7＝指令码。

② 读数据：E＝H，RS＝H，RW＝H，数据从 D0～D7 得到。

③ 写数据：E＝高脉冲，RS＝H，RW=L，D0～D7＝数据。

(4) LCM1602 液晶模块初始化过程

① 延时 15ms。

② 写指令 38H，设置显示模式。

③ 延时 5ms。

④ 写指令 38H，设置显示模式。

⑤ 延时 5ms。

⑥ 写指令 38H，设置显示模式。

⑦ 写指令 06H，显示光标移动设置。

⑧ 写指令 0CH，显示开及光标设置。

⑨ 写指令 01H，显示清屏。

⑩ 写指令 80H，定位数据指针到 80H。

(5) LCM1602 数据 RAM 读写

① 读显示RAM

首先，写指令码(指令码为目标RAM地址)。然后，从D0～D7口获得数据。

② 写显示RAM

首先，写指令码(指令码为目标RAM地址)。然后，写数据到D0～D7。

(6) 数据控制

液晶控制器内部设有数据地址指针。

3.6 硬件 看门狗 电路

在MSP430单片机使用过程中，由于所工作的环境会有电源波动等随机的干扰产生，为了使单片机工作可靠，在出现电源波动的情况下，要使MSP430可靠复位，因此必须
外加硬件看门狗电路。

本系统采用IMP706R看门狗芯片。该芯片功耗低、使用简单、价格低廉、可靠性高，应用十分广泛。

该芯片将在以下情况产生复位信号：

① 该芯片当电源电压低于3.08V将会产生一个复位信号。

② 在系统上电、掉电和电源中断将产生一个复位信号。

③ 如果WDI引脚在1.6秒以内没有脉冲清零操作，同样会产生一个复位信号。该芯片内部集成一个电源监测电路。

硬件封装图如图3.11所示。

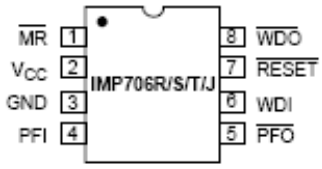

图 3.11 IMP706R 封装

Fig. 3.11 Packge of IMP706R

其中，MR：手动复位输入端；Vcc：+5V 电源输入端；GND：公共地端；PEI：电源故障电压监控输入端，当 PEI 小于 1.25V 时，PEO 输出为低电平；PEO：电源故障输出端；WDI：“看门狗”输入端；RESET：复位输出端，低电平有效；WDO：“看门狗”输出端。

在系统中的应用电路图如图 3.12 所示。在本开发板中未设计硬件看门狗，如果设计产品，请将硬件看门狗添加上，因为 MSP430 内部的看门狗在电源波动的时候会受影响。

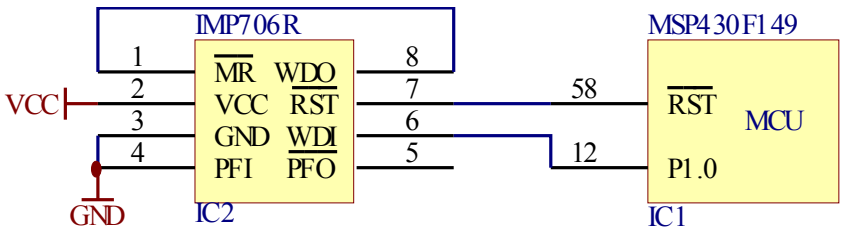

图 3.12 IMP706R 应用连接图

Fig. 3.12 Application connection chart of IMP706R

3.7 嵌入式网络模块其他电路设计

(1) 电源部分

由于嵌入式网络模块的功耗极低，故电源部分设计考虑到成本和稳定性选择了普通变压器再整流稳压的电路。稳压芯片为 LM17-33，为 3.3V 的稳压芯片。电路中在每个芯片电源引脚附近都设计了去耦电容，保证了电源无干扰。

(2) 接口部分

液晶属于可选择安装的部件，因此在 PCB 制板图上留出了接口，如果需要安装，通过排线连接即可。

另外还开扩充了串行接口，在 PCB 制版图留出了位置，方便功能扩展。

JTAG 采用 IDC14 接插件。

以上讲述了嵌入式 TCP/IP 的基本原理和精简设计准则，本开发板的 TCPIP 协议已经提供了 API 应用接口函数，应用中只需调用接口函数即可。请仔细阅读源代码，详细的注视帮助你理解整个嵌入式 TCPIP。

参考文献

[1] MSP430F149 SIGNAL MICROCONTROLLER Data Sheet. Texas Instruments Inc.,2004.

[2] CS8900A Product Data Sheet. CIRRUS LOGIC Inc.,1999.

[3] 范建华等译.[美]史蒂文斯（W.Richard Stevens）著 TCP/IP 详解,卷 1:协议.机械工业出版社, 2000.

[4] 周明天,汪文勇.TCP/IP 网络原理与技术.北京:清华大学出版社,1993.

[5] 王勇,陈抗生.嵌入式 Internet 中的协议选择.电信科学,2002,(4):15-18.

[6] 王勇,姚亦峰.嵌入式系统接入 Internet 的技术研究.计算机工程与应用,2001,37(4):16-19.

[7]王勇,姚亦峰,陈抗生.一种嵌入式系统接入 Internet 的方法及现.电子技术,2000,27(9):12-15.

[8] 吕京建,肖海桥.嵌入式 Internet 技术及其应用.http://www.21ic.com,2002.

[9] 牟英峰,徐殿国,张东来.基于嵌入式 TCP/IP 协议栈的信息家电连接 Internet 单芯片解决方案. 电子技术应用,2002,28(6):16-18.

[10] 胡大可.MSP430 系列单片机 C 语言程序设计与开发.北京:北京航天航空大学出版,2003.