

基于 MC9S12NE64 的嵌入式 Web 服务器的研究与实现

Research and Implementation of Embedded Web Server Based on MC9S12NE64

赵仕俊 李素侠 (石油大学(华东)计算机与通信工程学院, 山东 东营 257061)

摘要:针对石油仪器中大量存在的低端嵌入式设备的特点,提出了一种利用 MC9S12NE64 构造嵌入式 Web 服务器作为网络接入模块的单芯片方案。阐述了实现嵌入式 Web 服务器的关键技术—HTTP 协议和通用网关接口 CGI。

关键词:嵌入式 Web 服务器 通用网关接口(CGI) 超文本传输协议(HTTP) 石油仪器 单芯片

1 引言

工控设备、信息家电和个人通讯产品等嵌入式系统的智能化、网络化越来越要求具有网络功能;企业的信息化改造也需要将大量的非网络设备接入 Internet,并且通过 WEB 浏览器对这些设备进行监控和统一管理。如何将 Internet 技术应用到嵌入式设备中,实现对嵌入式设备的远程控制,成为业界关注的焦点。

笔者从事研究的工作主要是石油仪器仪表的研制与生产,现在的每台仪器都是独立的,无法通过 Internet 进行远程测量控制和动态信息的发布。随着 Internet 的普及,越来越多的设备都要求接入 Internet 进行信息共享。所以有必要对现有的石油仪器仪表增加网络功能,以实现信息共享。而且实现了网络化后,通过 Internet 用户可以远程对仪器进行功能操作、获取测量结果并对仪器实时监控、设置参数和故障诊断,控制其在因特网上动态发布信息。

2 设计思想

目前,在嵌入式系统中实现 Web 服务器的应用方案很多,因为嵌入式系统提供了这项服务后,就为外部的信息采集与交互提供了一个公开的标准接口,可以通过现有的 Internet 网络将嵌入式系统所提供的服务延伸出去,方便对嵌入式设备的远程控制和查询。

嵌入式系统中 Web 服务器的实现方法主要有两种:目标设备上有操作系统;目标设备上无操作系统。

前者适用于高端的嵌入式设备,这类设备资源相

对丰富,有足够的资源可供移植嵌入式操作系统,这类设备要实现的 Web 功能相对比较复杂。后者适用于低端设备,它对嵌入式设备的硬件要求比较低。低端设备上要实现的 Web 服务器一般功能比较简单,不需要操作系统提供复杂的功能支持。

由于现在的石油仪器中大量存在的是 8 位或 16 位单片机,本方案的设计思想就是基于后者,也就是在无操作系统的 16 位单片机中实现一个嵌入式 Web 服务器。

3 硬件平台

嵌入式系统硬件包括微处理器、存储器、外设器件和 I/O 端口等,其核心是嵌入式微控制器。为适应上网需求,嵌入式微控制器不仅要具有传统的控制功能,还要有与 Internet 连接的功能。我们选用 Freescale 公司的 16 位单片机 MC9S12NE64 芯片,该嵌入式设备配备了 64K Flash, 8K RAM。该芯片最大的特点是内部集成以太网控制器,并且提供了以太网连接所需要的一切:通信堆栈、以太网媒体访问控制器(EMAC)、10/100 以太网物理层(EPHY)和片上闪存。其硬件结构框图如图 1 所示。

MC9S12NE64 提供了一种单芯片的解决方案,它内置有 CPU、FLASH、RAM、MAC 和 PHY,因为不需要外部器件的连接,减少了嵌入式设备实现以太网连接的成本和复杂性。虽然如此,要实现以太网的接入仍然需要一些外围接口部件,如 25 - MHz 的晶振,3.3V 的电源,连接 PHY_RBIAS 引脚的电阻,RJ45 接头,高速 LAN 电磁隔离器,电容等。

由于 MC9S12NE64 内置了 MAC 和 PHY, 所以连接一个 RJ45 接口非常容易, 需要注意的是在 MC9S12NE64 和 RJ45 之间必须安装一个高速 LAN 电磁隔离器。图 2 为以太网接口电路图。

协议栈, 它根据嵌入式系统的特点裁减了标准的 TCP/IP, 能够正确接收和发送 ARP、IP、ICMP 和 TCP 报文, 为应用层提供底层支持。针对低端嵌入式设备资源有限的特点, 在设计中对 HTTP 协议和 CGI 进行了简化。

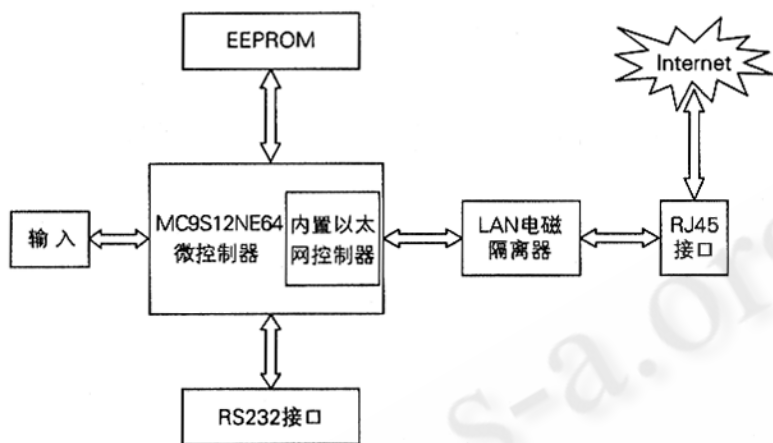


图 1 嵌入式 Web 服务器硬件结构框图

4.1 HTTP 协议的实现

用 HTTP 协议实现的 Web 服务器能够接收客户的请求数据包、读取请求消息、进行解析处理并将相应的处理结果发送给客户端。一个简单的 HTTP 服务模型如图 3 所示。

4.1.1 解析请求数据报文

当 Web 客户端访问 Web 服务器时, 首先向服务器发出请求, 服务器收到请求后, 对请求数据报文(其格式如图 4 所示)进行解析。本文的设计方案中, Web 服务器仅接受 GET 和 POST 两种请求方法, 其中 GET 方法用于客户端获取页面文件, 而 POST 方法用于客户端向 Web 服务器的设备发送控制信息。HTTP1.

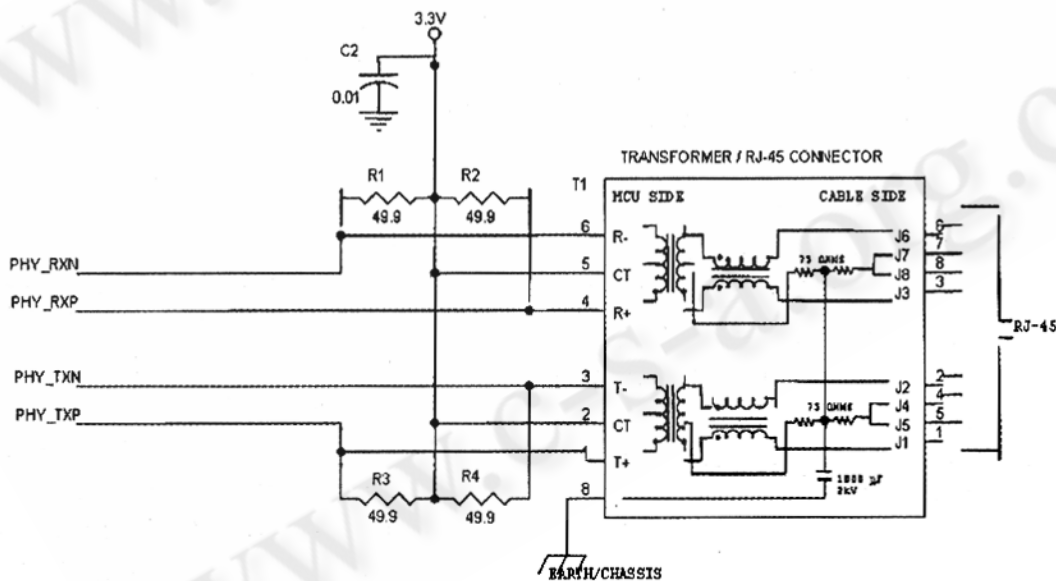


图 2 以太网接口电路图

表 1 描述了 MCU 与高速 LAN 电磁隔离器的信号线连接。

4 关键技术

实现嵌入式 Web 服务器的关键技术是 HTTP 协议和通用网关接口 CGI。本设计在开发板上移植了 OpenTcp 协议, 这是一个适合 8/16 位单片机的 TCP/IP

1 中定义的一些其他方法由于在设计中作用不大而不做处理。元信息的结构相当复杂, 在本设计中处理与否对系统性能影响很小, 也不做处理。

当请求数据报文到达时, 先解析是否为 GET 或 POST 方法, 否则不作处理。如果是 GET 方法, 表明客户端请求一个页面, 继续解析出页面的名称, 送出应答数据报文。如果是 POST 方法, 提取出 POST 提交的信

息,控制 WEB 服务器端的设备的硬件部分。

4.1.2 封装应答数据报文

应答数据报文(格式如图 5 所示)的协议版本号、应答状态码、应答状态码说明是一些相对固定的信息,接收的元信息部分的数据包括服务器类型、应答的 GMT(格林尼治标准时间)、内容类型、内容长度、是否使用 Cookie 等,这些参数不是必需的,可以只给出部分确定的信息(如服务类型、内容类型等)。所有应答 HTTP 报文的头部都可以固定不变地存储到存放 Web 页面的 Flash 中。

表 1 高速 LAN 电磁隔离器连接电路

| 高速 LAN 电磁隔离器 | MC9S12NE64 引脚 |
|--------------|---------------|
| TX CT | 3.3V |
| T + | PHY_TXP |
| T - | PHY_TXN |
| RX CT | 3.3V |
| R + | PHY_RXP |
| R - | PHY_RXN |

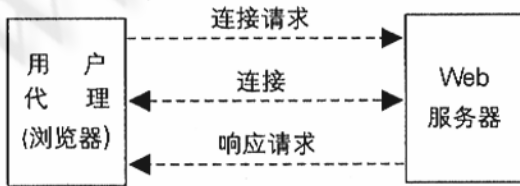


图 3 HTTP 服务模型

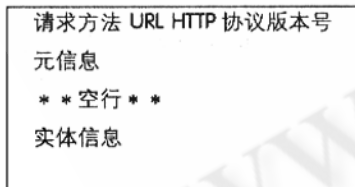


图 4 请求数据报文的格式

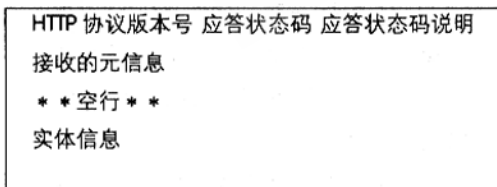


图 5 应答数据报文的格式

如果 Web 服务器接收到的请求 HTTP 报文使用的是 GET 方法,表明客户端要求获得一个页面,则服务器开始解析所请求页面的 URL。传统的文件系统是按名字查找和存放,由于文件名千差万别,如果使用字符串匹配的方法进行解析将造成文件处理缓慢,影响文件系统效率。这里我们采用了字符串 HASH 变换,在 EEPROM 中存放文件名的 HASH 值,将复杂的字符串比较变为简单的 HASH 值比较,极大地提高了文件查找、读写速度,减小了系统的响应时间。

4.2 CGI 的实现

4.2.1 CGI 的工作流程

由于嵌入式 Web 服务器的应用领域是专门的控制设备,控制这些设备的流程都是基本固定的,所以只要能提供某一种交互功能的 Web 服务器就完全可以适用于控制设备。又由于 16 位 MCU 系统的局限性,使得软件不能涉及进程管理、动态内存分配等概念,标准的 CGI 在这种环境下无法完全实现。该 Web 服务器只提供单纯的 CGI 功能,使用户能够跟设备进行必要的交互,对 Web 服务器的其它功能则不支持。对于原先标准 CGI 作为独立可执行文件完成的功能,移到 Web 服务器内部作为一个函数处理,给用户留有必要的接口。

Web 服务器接收浏览器请求进行分析,判断如果没有从客户端传来参数,则调用用户的 get_state() 函数取得设备的当前状态显示给客户端浏览器,用户通过操作界面设置设备的新参数,发回给 Web 服务器,Web 服务器利用 CGI 处理模块分解出传来的参数,并通过用户提供的 get_state() 函数,设置设备的新状态,同时返回给客户端。客户端浏览器可以和 Web 服务器进行连续交互,以此监控设备的运行。其工作流程如图 6 所示。

4.2.2 主要函数分析

```
void do_service( int sd)
```

该函数是主要处理函数,完成读请求,分析请求,把请求中传来的参数送给 CGI 处理函数,把获取的机器状态或用户设置的机器新状态封装成 HTML 格式送回给浏览器。

```
Request_line * parse_request_line( request_line * r);
```

分解请求行,把接收到的请求行分解为请求方式、URL、HTTP 版本号三部分。

```
Request_line * handle_filename(request_line * r);
```

获取 GET 方式时“?”之后的数据。

```
Header_rec * parse_header(char * array, header_rec * header, char c)
```

请求分解为名字、值两部分,分别是 header -> name = Content_length; header -> value = 58。

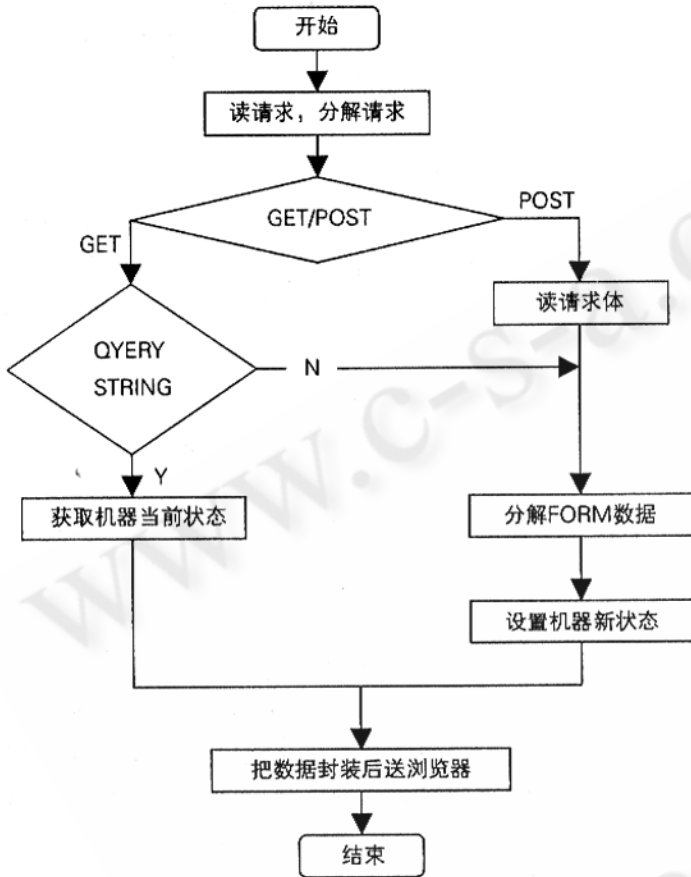


图 6 流程示意图

```
Void cgi_process(char * input);
```

分解 FORM 提交的数据,把标准 CGI 脚本所做的工作引入 WEB 服务器程序中。

```
Void pack_html_formation(char * form_cgi[]);
```

用来把发送回浏览器的系统状态数据封装成 HTML 格式。由 do_service 发送

```
void send_status_line(char * array, int sd);
```

发送状态行,对于单片机 WEB,简化为发送一个固定的状态行"HTTP/1.1 200 OK\r\n"。

```
void send_white_line(int sd);
```

因为这里响应的头被省略了,所以发送空白行,标志响应消息头的结束。

4.2.3 用户接口

CGI 部分放入服务器后,用户通过接口函数就可以控制具体的仪器:

```
Void get_status(char * stat_list[])
```

用来获取仪器的当前状态,由用户根据具体的仪器来完成。可以把获取到的各个控制选项值转换为字符型按固定顺序放入数组 stat_list[]。

```
Void set_status(char * stat_list[])
```

从 stat_list[] 中取传来的新参数设置控制仪器中相应选项来控制仪器的运行。

```
Void set_pid(char * pid_list[])
```

该函数用来设置仪器的相关参数。

5 结束语

本文设计并实现了一个基于 MC9S12NE64 的小型嵌入式 Web 服务器,为石油仪器提供了一种网络化接口。硬件设计只需要电源、电阻、晶振等基本外围电路,避免了由于大量的连接而产生的可靠性问题。根据系统要求简化了 HTTP 协议和 CGI 的一些机制,在资源有限的情况下设计满足需求的嵌入式 Web 服务器。

参考文献

- 1 MC9S12NE64 Data Sheet, <http://www.freescale.com/>, 2004,9.
- 2 MC9S12NE64 在线座谈, <http://www.freescale.com.cn/products/>.
- 3 Freescale Application Note 2836. Web Server Development with MC9S12NE64 and OpenTCP. 2004, 9.
- 4 张曦煌,基于嵌入式 Web 的远程仪表接口[J],微计算机信息,2006,22(1):147-149.
- 5 高珀珀、邵时,低端嵌入式设备 Web 服务器的研究与实现[J],计算机工程,2005,31(5):219-221.
- 6 张曦煌、柴志雷,嵌入式设备 Web 服务器中 CGI 的特点与实现[J],小型微型计算机系统,2003,24(11):2046-2048.
- 7 陈丽娟、白瑞林,基于 IP2022 单芯片的嵌入式 Web 服务器设计[J],江南大学学报(自然科学版),2005,4(4):348-351.