

# 气象环境监测系统中嵌入式 Web 服务器<sup>①</sup>

梁永恩<sup>1</sup>, 万世明<sup>2</sup>, 汤海林<sup>1</sup>

<sup>1</sup>(广东白云学院 计算机系, 广州 510450)

<sup>2</sup>(武汉软件工程职业学院 计算机学院, 武汉 430205)

**摘要:** 为了提高气象环境监测系统的响应速度和运行效率, 满足监测中心实时、准确地监测气象环境信息的要求, 设计并实现了基于 GoAhead 与 Ajax(asynchronous JavaScript and XML)的嵌入式 web 服务器. 通过 CGI(Common Gateway Interface, 通用网关接口)程序实现了对环境数据的动态更新. 该系统操作简单, 运行稳定, 响应快速, 可以远距离监测, 有助于管理部门及时了解被测地点的温湿度、风向等气象信息.

**关键词:** GoAhead; Ajax; 嵌入式 web 服务器; 通用网关接口

## Embedded Web Server in Meteorological Environment Monitoring System

LIANG Yong-En<sup>1</sup>, WAN Shi-Ming<sup>2</sup>, TANG Hai-Lin<sup>1</sup>

<sup>1</sup>(Dept. of Computer Science, Guangdong Baiyun University, Guangzhou 510450, China)

<sup>2</sup>(Dept. of Computer Science, Wuhan Vocational College of Software and Engineering, Wuhan 430205, China)

**Abstract:** In order to improve the response speed and efficiency of meteorological environment monitoring system, and meet the requirements of getting meteorological information in real time and accurately for the monitoring center, an embedded web server based on GoAhead and Ajax (asynchronous JavaScript and XML) is designed. In this system, environmental data is dynamically updated by using CGI(Common Gateway Interface). The system has the advantages of simple operation, stable running, rapid response and long distance monitoring, which is helpful for management to know the temperature, humidity, wind direction and other information of the observation posts.

**Key words:** GoAhead; Ajax; embedded web server; common gateway interface

随着嵌入式技术的发展, 嵌入式终端设备越来越多地应用在环境监测、交通管理、汽车电子设备、工业控制、信息家电、医疗设备、移动计算设备等多个领域. 将嵌入式 Web 服务器引入嵌入式设备中, 使传统的嵌入式设备转变为以 TCP/IP 为底层通信协议, 可通过互联网实现远程监控的智能设备<sup>[1-3]</sup>. 这样用户可以通过 PC、手机和平板电脑等终端设备使用浏览器作为接口访问嵌入式设备, 大大提高嵌入式设备的易用性.

嵌入式系统受到处理器能力和存储空间限制, 可利用的资源比 PC 少得多, 并且也不需要能同时处理很多用户的请求, 因此不会使用 Linux 下最常用的如 Apache(Apache HTTP Server) 等服务器, 而需要使用一些专门为嵌入式设备设计的 Web 服务器. 传统的网页如果需要更新内容, 必须重新加载整个网页, 这

将会大大加重服务器端的负载. 使用 Ajax 技术<sup>[4,5]</sup>, 将 JavaScript 程序内嵌在网页中, 不需要任何浏览器插件, 通过在后台与服务器进行少量数据交换, 使网页实现异步更新. 同时, 为实现在客户端和服务器程序之间传输数据, 本文采用 CGI 技术, 与 Ajax 的 XMLHttpRequest 组件进行交互, 实现了动态 Web 功能. 利用 Ajax 及 CGI, 设计了气象环境嵌入式 web 服务器, 可以实现对气象信息的实时监控和记录查询等功能.

## 1 硬件设计

系统硬件由 S3C2440 处理器、DRAM、Nand Flash、JTAG、RJ45 接口、温湿度传感器、风速风向传感器等部分组成, 如图 1 所示. CPU 选用三星公司的 S3C2440, 利用 S3C2440 实现对气象数据的实时检测, 同时通过

①收稿时间:2013-12-17;收到修改稿时间:2014-01-20

RJ45 网卡接口对数据进行远程传输. S3C2440<sup>[6]</sup> 基于 ARM920T 核心, 具有丰富的系统接口, 如多种串行接口(SPI、UART、IIC、IIS 等)、130 个通用 I/O 口、8 通道 10 位 ADC、SD 接口及 MMC 接口等, 功耗低, 适合于对成本和功率敏感型的应用.

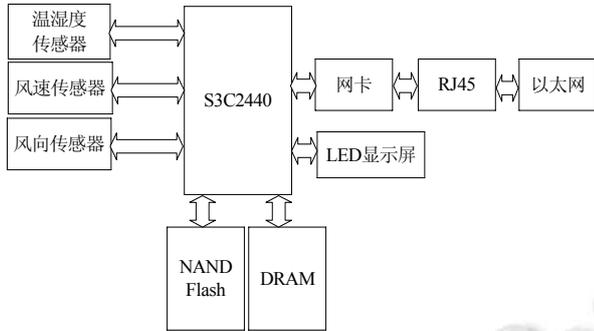


图 1 系统硬件结构

## 2 软件架构

系统的软件架构如图 2 所示. 最底层是设备驱动层, 该层直接与硬件打交道, 为上层软件提供了设备的操作接口, 以太网驱动、串口驱动、ADC 驱动等设备驱动程序都在这一层. 设备驱动层之上是操作系统层, 包括文件系统、操作系统内核、TCP/IP 网络系统等<sup>[7,8]</sup>. 操作系统层之上是用户的应用软件层, 本系统主要的应用程序是 web 服务器.

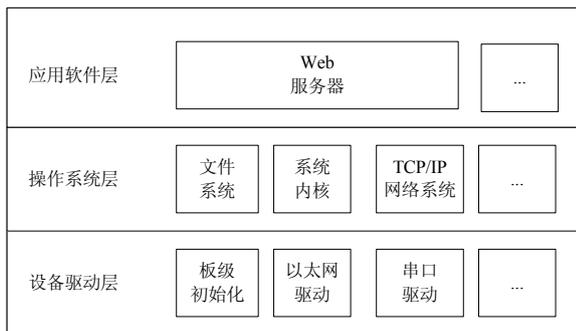


图 2 软件架构

这里操作系统采用开源的嵌入式 Linux 系统, 内核版本为 Linux 2.6.32.2, 交叉编译器为 arm-linux-gcc-4.4.3. 将嵌入式 Web 服务器通过交叉编译为嵌入式平台的版本, 然后将该程序以操作系统的的一个子进程在后台运行, 用户可以通过 IE 等浏览器访问设计好的网页, 从而获取气象数据.

## 3 嵌入式 Web 服务器的原理与实现

### 3.1 嵌入式 Web server 与移植

由于在嵌入式设备中的存储资源和运算能力有限, 因此嵌入式设备中的 web 服务器只需提供必要的功能即可. 一般嵌入式 web 服务器的大小在几十 KB 左右. 常见的嵌入式小型 web 服务器有: GoAhead、Boa、Lighttpd、Shttpd、Thttpd、Mathopd、Mini\_httpd、Appweb<sup>[9,10]</sup>. GoAhead 是一款强大的免费的嵌入式 web 服务器, 广泛应用在各种嵌入式系统中. 支持各种多种操作系统如 VxWorks、Linux、Lynx、windos 等. 可以使用静态 html, cgi 或 ASP(Active Server Page, 动态服务器页面)以及嵌入式的 JavaScript. 其移植过程如下:

1) 下载 goahead 的源码压缩包并解压到 PC 机 Linux 的任一目录.

2) 修改 goahead 的 mkfile 文件. 打开 goahead/LINUX/Makefile 文件, 修改 gcc 和 ar 变量, 如下两行:

```
CC=/usr/local/arm-gcc/opt/toolschain/4.4.3/bin/arm-linux-gcc
```

```
AR=/usr/local/arm-gcc/opt/toolschain/4.4.3/bin/arm-linux-ar
```

3) 由于 gethostbyname() 函数在开发板上无效, 需将 goahead/LINUX/main.c 的 initWebs 函数中的如下代码注释:

```
if ((hp = gethostbyname(host)) == NULL) {
    error(E_L, E_LOG, T("Can't get host address"));
    fprintf(stderr, "initWebs: host name %s\r", host);
    return -1;}
memcpy((char *) &intaddr, (char *)
hp->h_addr_list[0], (size_t) hp->h_length);
```

4) 编译. 在 goahead/LINUX 目录下执行 make 命令进行编译, 在此目录下生产 webs 可执行文件.

### 3.2 动态交互的实现

传统 Web 应用使用的是同步交互方式, 用户操作触发一个 Http 请求, 服务器接收用户请求进行数据处理, 然后返回一个新的网页. 由于整个数据的处理过程均在服务器端完成, 且页面在每次提交数据后都会重新加载页面, 当处理时间较长时可能会造成浏览器在一段时间内无响应, 用户体验较差, 这大大增加了服务器的负担, 降低了网络传输效率. 应用 Ajax 技术, 客

户端可以在不更新加载整个页面的情况下与服务器交换数据并更新部分网页,从而可以较好的同步交互造成的响应慢的问题. Ajax 的核心对象是 XMLHttpRequest, 通过 XMLHttpRequest 对象, 浏览器的 JavaScript 可以后台与服务器交换数据, 无需等待服务器的响应. 因此在服务器和浏览器之间交换的数据大量减少, 大大提高了页面的响应速度.

CGI 是嵌入式 Web 服务器的核心内容之一, CGI 技术用于在服务器端发布网页, 与 XMLHttpRequest 组件进行交互, 实现动态 Web 功能<sup>[1]</sup>. 通过 CGI 及 XMLHttpRequest, 可以动态获取嵌入式设备的数据(如本系统中的温湿度、风速风向等信息)并动态更新网页. 采用 AJAX 和 CGI 的嵌入式 Web 服务器的系统模型如图 3 所示.

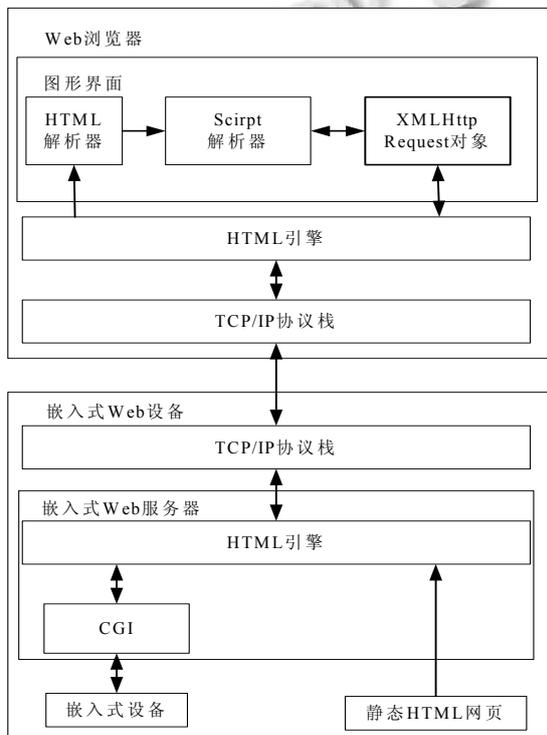


图 3 采用 Ajax 和 CGI 的嵌入式 Web 服务器的模型

下面以系统中更新气象数据为例介绍动态交互的具体实现步骤:

1) 创建 HTML 页面.

可以直接用 HTML 语言编写 HTML 页面, 也可通过 Dreamweaver、FrontPage 等软件创建页面. 部分代码如下. 该 HTML 文件使用了一个名为“xmlhttpreq.js”的 Javascript 脚本文件.

```

<html>
  <head>
    <meta http-equiv="Content-Type"
content="text/html; charset=UTF-8"/>
    <script language="JavaScript"
src="xmlhttpreq.js"></script>
  </head>
  <body>
    .....
    <p> 温度 : <div id="current_temperature
"></div></p>
    <p> 湿度 : <div id="current_ humidity
"></div></p>
    .....
    <input type="button" value=" 更 新 "
onclick="sender()" />
  </body>
</html>

```

程序包含多个 div 和一个更新数据的按钮. div 部分用于显示来自服务器的信息如温度、湿度等. 当按钮被点击时, 它负责调用名为 sender() 的请求更新数据的函数, sender() 函数定义放在 Javascript 脚本文件中.

2) 建立一个 Javascript 脚本文件, 实现了 Ajax 进行异步访问服务器.

为实现结构和行为的分离, 本系统将 JavaScript 写成独立的文件. 在该文件中, 包括了创建 XMLHttpRequest 对象的函数、发送更新气象数据的请求函数、回调处理函数等.

实现 JavaScript 脚本程序的关键代码如下.

```

var xmlhttp;
function createXHR()
{
  if (typeof XMLHttpRequest != "undefined")
  { xmlhttp = new XMLHttpRequest();}
  else if (window.ActiveXObject)
  { var aVersions = ["Msxml2.XMLHttp.5.0",
"Msxml2.XMLHttp.4.0", "Msxml2.XMLHttp.3.0",
"Msxml2.XMLHttp", "Microsoft.XMLHttp"];
  for (var i = 0; i < aVersions.length; i++) {
    try {xmlhttp = new

```

```

ActiveXObject(aVersions[i]);
        return xmlhttp; }
    catch (e) {} }
}

function sender()
{   xmlhttp = createXHR();
    if(xmlhttp)
        {xmlhttp.onreadystatechange=
updateFunction;
        xmlhttp.open("GET",
"cgi-bin/Updatedata.cgi?cur_temperature = " +
Math.random());
        xmlhttp.send(null); } } }

function updateFunction()
{   if (xmlhttp.readyState == 4)
    { if (xmlhttp.status == 200)
        {var returnValue = xmlhttp.responseXML;
        var txt ="";
        if(returnValue != null && returnValue.length > 0)
            { //对返回的数据处理并更新页面
            .....
            document.getElementById("current_temperature ").
innerHTML = txt; ..... } } } }

```

createXHR() 函数用于创建异步访问对象 XMLHttpRequest, 这里根据不同的浏览器设置不同的创建方法。sender()函数由 HTML 文件调用, 作用是提交异步请求。在该函数中, 首先调用 createXHR()函数创建一个 XMLHttpRequest 对象, 然后调用 XMLHttpRequest 对象的 open()函数规定请求的类型、URL 以及是否异步处理请求, 最后调用 send()方法将请求发送到 web 服务器中, 此处的 URL 是服务器上的 CGI 应用程序, 在 Updatedata.cgi 后面跟个 cur\_temperature 参数是为了防止 Ajax 页面缓存。当 XMLHttpRequest 对象的 readyState 改变时, 会触发 onreadystatechange 事件。我们将 onreadystatechange 属性设置为 updateFunction()函数。updateFunction()为异步回调函数处理, 在 readyState 改变会自动调用该函数。在 updateFunction(), 当响应已就绪(此时 readyState 等于 4 且状态为 200)时处理返回的数据

并更新页面。

### 3)建立服务器端应用程序

cgi 程序可以用 C,C++,perl 和 VB 等高级语言编写, 嵌入式系统往往需要对嵌入式硬件设备访问和控制, 故在嵌入式系统中用 C 语言编写 cgi 程序比较多。Updatedata.c 的部分内容如下, 该程序的功能是向浏览器发送包含气象信息的 XML 文档。

```

int main(void)
{
    float temperature, humidity, windspeed;
    int winddirection;
    printf("Content-type: text / xml; charset = utf-8 \r\n\r\n");
    //获取气象信息并以 XML 的形式写入到 XML 文件中
    .....
    printf(气象信息);
}

```

程序编写后, 进行交叉编译, 在命令行输入“arm-linux-gcc -o Updatedata.cgi Updatedata.c”生成 Updatedata.cgi 文件。

## 4 测试

将网页文件、Javascript 脚本文件下载到嵌入式设备的 Goahead 服务器的文档根目录, Updatedata.cgi 放在 Goahead 服务器的 cgi 配置目录。启动服务器程序后, 在接入网段的 PC 机上打开浏览器窗口, 输入嵌入式 web 服务器的 IP 地址, 即可进入气象环境监测系统页面, 如图 4 所示。



图 4 气象环境监测系统页面

## 5 结语

本文结合 AJAX 和 CGI 技术,设计并实现了基于 GoAhead 的气象环境监测嵌入式 Web 服务器。通过 AJAX 技术和 CGI 实现了 web 异步动态交互,有效减少了服务器的负担。客户端无需安装软件,用户通过浏览器就能直接监测被测点的气象环境信息。应用表明,该系统工作稳定可靠,人机交互良好。

### 参考文献

- 1 谢华成,李晶.物联网系统中嵌入式 BOA 的移植与应用.软件,2011,32(5):106-108.
- 2 王向群,徐沛平,渠毅等.基于 GoAhead webservice 的应用开发.计算机系统应用,2011,20(12):221-223.
- 3 丁苍峰,薛宁静,吕茂林.嵌入式 Web 服务器的设计研究.计算机工程与设计,2009,30(18):4188-4191.
- 4 吴灿培,胡顺豪,王海航,等.基于 Ajax 和 SVG 的 Web 远程实时监控系统的实现.计算机工程与设计,2011,32(9):3004-3007.
- 5 潘琢金,王秋实.嵌入式 Web 服务器中动态 Web 技术的研究.计算机工程设计,2010,31(18):3975-3978.
- 6 韦东山.嵌入式 Linux 应用开发完全手册.北京:人民邮电出版社,2008.
- 7 俞辉.ARM 嵌入式 Linux 系统设计与开发.北京:机械工业出版社,2010.
- 8 文全刚.嵌入式 Linux 操作系统原理与应用.北京:北京航空航天大学出版社,2011.
- 9 赵庆磊,韩诚山,文明,等.GoAhead 嵌入式服务器的设计与实现.微计算机信息,2011,27(4):67-69.
- 10 周彬,秦玉娟,刘馨.基于 GoAhead 的油井多参量远程监测系统的设计.低压电器,2013(11):23-26.
- 11 李勇.CGI 在嵌入式 Web 服务器中的应用和实现.微计算机信息,2008,24(10):110-111.