

嵌入式系统网络互连技术的设计与实现

Research into Embedded Micro Internetworking Technology and its Implementation

陈利锋¹ 刘博勤² 涂时亮¹ 陈章龙¹

(复旦大学微机实验室 上海200433)¹

(西南师范大学计算机与信息科学学院 重庆400715)²

Abstract Based on the existing Internet technology, this article introduces some new technologies concerning the connection between embedded devices and Internet, which are designed for electronic devices. The prospects for the application of this technology are proved to be good and vast.

Keywords Embedded system, Singlechip, Gateway, Serial communication, ActiveX control, Socket

由于网络技术的成熟与完善和互联网络的迅速发展,许多嵌入式设备如电视、电话、数控机床、PDA(个人数字助理),也有了网络接入的功能需求,促进了嵌入式网络设备和产品的开发生产,这些设备和产品正在改变人们的生活方式,也给生产和人们的生活带来了极大的便利。我们可以设想:若是能够将我们日常生活里的设备连到因特网,比如说,用手机通过远程遥控来打开家里的电热水器或空调(用手机收发 eMail 技术已经逐渐成熟),或者维修员在维修客户家中的空调之前,就已先准确地弄清楚空调故障的部位和需要更换的零件,从而能够有备而去,在工业生产实践中实现对工程设备的实时监控,进行远程数据采集、数据处理以及控制信号的产生与传输等远程控制……。这都是非常诱人的,那么,能否开发一种技术,既能充分利用现有的 Internet 技术和资源,又能以很低的成本将它应用到各类智能设备和家用电器中去呢?

1 引言

随着信息电器(IA)的出现和蓬勃发展,对电器的智能性要求越来越高,智能电器的核心部件是嵌入式芯片。以单片机为核心的嵌入式系统,可以实现数据采集、自动控制等功能,并且具有体积小,功耗低,价格低廉,可操作性能良好,应用环境广泛等优点,在生产生活中得到广泛的应用。随着现代信息技术的发展以及计算机网络的广泛使用,计算机通信技术已经日臻成熟,有许多方式可以实现此类嵌入式系统的网络接入,如,RF Modem, IrDA, NetCard, USB, RS232 RS485等等。其中串行通信作为一种灵活方便可靠的通信方式,仍不失为有效的通信手段,被广泛应用于工业控制中。

在工业生产实践中,对工程实现实时监控,通常要求能在用户界面上具有数据采集、数据处理以及控制信号的产生与传输等功能,这些通信都可以通过串行通信来进行。如果加上网关连接,将嵌入式设备连接到网关,由网关通过 Internet, Intranet, LAN, WAN 等多种方法来将嵌入式系统接入到网络,采集到的数据和向其发送的控制命令等信息通过网络来传输,可实现对设备的远程控制。

与其他的嵌入式系统网络接入技术比较,这种通过串行连接的方式的优点在于,不需要在嵌入式系统上有本地操作系统(Native OS),也不需要在此系统上实现 TCP/IP 等网络通信协议,由此大大降低了对硬件的需求,只要能实现串行口通信就可以实现网络接入,使其可以应用在更广泛的领域。

在 Windows 环境下,用 VC++ 6.0 作为开发工具,可以很方便地进行串口通讯应用程序的开发,实现对串口通讯的精确控制。Visual C++ 6.0 里的 Active X 控件 Microsoft Communications Control version 6.0 封装了控制串口通讯所需的各种参数和方法,用它来进行串口通讯程序设计,可以避免对硬件进行参数设置和通信控制的复杂操作。利用 Socket 来进行网络通信的开发,可以根据需要采用不同的通信方式,提供不同等级的服务质量(QoS),为通信的顺利进行提供了强大的支持和极大的灵活性。

2 嵌入式系统网络互连的体系结构及协议

2.1 嵌入式系统网络互连的体系结构

嵌入式设备通过串口连接在网关上,嵌入式设备

陈利锋 硕士研究生,研究方向 嵌入式系统与网络技术;刘博勤 副教授,访问学者;涂时亮 教授,博士生导师;陈章龙 教授。

与网关之间通过串行连接来进行通信。网关是可提供远程连接的机器，在工业控制上可使用卡式 PC，在智能家庭网络中则可为家庭智能控制器。网关一方面起到集线器的作用，将多个嵌入式设备连接到一起，便于集中管理；另一方面还起到协议转换的作用，将信息在内部串口通信协议和外部网络通信协议之间进行转换，保证可靠安全传输。网关接受控制台的请求，把控制台的控制命令转换成对嵌入式系统的控制命令，使嵌入式系统执行相应的动作；同时读取控制台所需的关于嵌入式设备的信息，通过外部网络反馈给控制台，

从而实现了对嵌入式设备的远程访问和控制。嵌入式系统网络互连的体系结构图如图1所示。

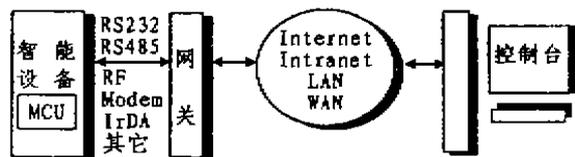


图1 嵌入式设备与 Internet 互连

表1 嵌入式设备和网关、网关和控制台之间的数据包名称清单

中文名称	英文名称	英文缩写
番号请求报文	Designation Requirement Datagram	DRD
核实口令请求 (设备给网关)	Verify Password Requirement To Gateway	VPRTG
核实口令请求 (网关给控制台)	Verify Password Requirement To Console	VPRTC
核实口令 (控制台给网关)	Verify Password To Gateway	VPTG
核实口令 (网关给设备)	Verify Password To Device	VPTA
拒绝控制 (设备给网关)	Refuse Control To Gateway	RCTG
拒绝控制 (网关给控制台)	Refuse Control To Console	RCTC
修改口令	Change Password	CPWD
口令修改成功	Password Change Succeed	PCS
口令修改失败	Password Change Fail	PCF
强制采集	Instant Collect	IC
执行指示 (控制台到网关)	Carry Out To Gateway	COTG
执行指示 (网关到设备)	Carry Out To Device	COTA
任命	Designation Datagram	DD
管理	Administrator Datagram	AD
移交管辖权	Hand Over	HO
	Ping	Ping
	PingA	PingA
设备属性表	Device Properties Description	APD
执行完毕 (网关到控制台)	Done To Console	DTC
执行完毕 (设备到网关)	Done To Gateway	DTG
监视	Monitor	MON
监视响应	Monitor Response	MR
CRC 错误	CRC Error	CRCE
采集报告 (网关到控制台)	Report To Console	RTC
采集报告 (设备到网关)	Report To Gateway	RTG
意外复位	Accident Reset	ARST
状态不明	Unknown State	UKS
登录请求	Login Requirement	LOGR
登录失败	Login Fail	LOGF
登录成功	Login Succeed	LOGS
串口不可得	Serial Port Unavailable	SPUAV
无法执行 (网关给控制台)	Fail	FAIL
没有权限	No Right	NRGHT
网关关闭	Gateway Quit	GQ
设备关闭	Device Quit	AQ
查询设备	Consult Device	CA

系统需要实现的功能包括：

- 嵌入式系统与网关之间的通信，可以采用 RS-232 的串行连接或 USB 的串口连接，或其他连接方式，由控制协议软件实现，根据需要实现差错控制，即插即用等功能；

- 网关对嵌入式设备进行控制，由于不同设备参数

的多少和意义不同，控制协议依设备而不同，相应的控制方式和命令也不同，网关对这些设备进行分类管理和相应的协议转换；

- 网关和远程控制台进行远程通信，响应控制台的请求，其中与嵌入式设备有关的请求要转换为相应的控制命令发送给嵌入式设备，所有控制信息的传送到

循相应的控制协议。

2.2 嵌入式系统网络互连的控制协议设计

对嵌入式设备进行控制必须遵照一定的协议来实现。对于每一类不同的嵌入式设备,它所能接受的命令和输出的信息是有很大的区别的,控制协议要完成的任务就是将这些命令和信息进行规范化,例如对于一个装有指示灯、传感器和动作部件的嵌入式设备来说,控制协议规定了该设备接受何种命令时打开指示灯,接受何种命令从传感器读取数据,接受何种命令时启动动作部件;当对该嵌入式设备进行查询时,所接受数据的含义如何理解。

另一方面远程控制台发来的请求必须转换为嵌入式设备能识别的命令,才能对该设备进行操作。这种转换可以通过称为驱动程序的中间层来实现,其功能就是将控制请求映射为对嵌入式设备的操作命令,这些功能的定义由网关和控制台之间的通信协议实现。

为了保证访问安全,必须在网关上实现授权机制。控制台根据网关的 IP 和端口号进行访问,而这两者对外是不公开的,控制台本身要经过认证过程才能登录到网关,这通过用户帐号制度实现。帐号的设立由网关管理员进行,远程控制台只能使用一授权的账号,而不能动态申请,更多的安全机制则涉及到加密和其他策略。

控制协议与具体的应用有关,其格式可以有多种实现方式,这里采用的是带标识符的数据包格式,具体形式为“命令+数据”。表1列出的是我们在实验过程中设计的一套命令体系,其中包括了一一般来说都必须具备的命令。

3 实现

我们在实验中以单片机作为嵌入式设备来模拟控制一些光电和机械部件(包括指示灯、报警器,电动马达,霍尔传感器等),以 PC 为代理网关,试验中成功地实现了通过 Internet 对嵌入式设备模拟器上的光电和机械部件的远程控制。

3.1 串行通信部分的实现

这里我们用 MCS-51 单片机代表嵌入式系统,网关作为代理网关,串行通讯接口电路如图2所示。

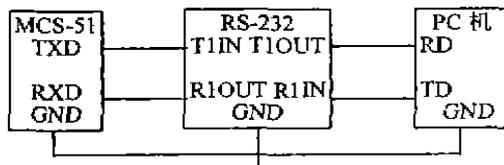


图2 串行通讯接口电路

嵌入式设备的串行通信功能一般是用汇编语言来

编写的,可以设置为四种工作方式:

- 移位寄存器输入输出方式,在串行口可以外接移位寄存器以扩展 I/O 接口,也可以外接输入输出设备。

- 8位的异步通信接口方式,传送一帧信息10位:1位起始位,8位数据位(先地位后高位),一位停止位,数据由 TXD 端输出,CPU 执行一条数据写入串行发送缓冲器 SBUF 的指令(如 MOV SBUF,A)将数据字节写入 SBUF 后,便启动串行口发送器进行发送,发送完一帧信息后就将发送中断标志位置位。

- 9位的异步通信接口方式,传送一帧信息11位:1位起始位,8位数据位(先地位后高位),一位可编程为1或0(附加的第9位数据),一位停止位,数据由 TXD 端输出,CPU 执行一条数据写入串行发送缓冲器 SBUF 的指令(如 MOV SBUF,A)将数据字节写入 SBUF 后,便启动串行口发送器进行发送,发送完一帧信息后就将发送中断标志位置位。

实验中我们采用了8位的异步通信接口方式,实现 MCS-51 单片机二进制缓冲输入方式的串行通信,汇编源程序请参见文[1]。

网关的串行通信:网关在进行串行通信之前必须对串行通信协议进行设置,串行通信有同步方式和异步方式两种,一般采用异步通信方式。在异步通信方式下,数据是以帧(Frame)的方式传送的,帧由起始位、数据位、停止位和校验位构成。串行通信可以采用查询方式或中断方式,为了实现实时控制,一般采用中断方式。如果采用数据块方式传送数据,还需要设置缓冲区,数据由中断处理例程放入缓冲区。在 Windows 环境下,可利用一个 Microsoft Communications Control 控件来实现对串口的通讯管理。

下面简要给出 PC 与嵌入式设备进行串行通信的程序:首先创建一个 MFC 应用程序,命名为 Gateway (网关)。在应用程序中插入 Microsoft Communications Control 控件,并再为其添加一个变量,命名为 m_serialComm。在程序的初始化函数中加入串口的初始化代码:

```
m_serialComm.SetCommPort(1); //选择 COM1
if(!m_serialComm.GetPortOpen())
    m_serialComm.SetPortOpen(TRUE); //打开串口
m_serialComm.SetSettings("9600,N,8,1"); //设置串口参数
//波特率9600bps,无奇偶校验,8位数据位,1位停止位
m_serialComm.SetInputMode(1); //设置二进制缓冲输入方式
m_serialComm.SetRThreshold(1); //每接受一个字符激发 OnComm 事件
}
```

当网关从串口接收到数据后,就激活一个 OnComm 事件,可以在 OnComm 消息处理函数中加入相应的处理代码,对数据进行处理。为控件添加处理 On-

Comm 事件的处理函数 OnOnCommMscomm(), 对接收的数据进行处理, 在这里所接收的数据是有一定格式的命令报文(“命令+数据”), 调用协议转换函数对数据报文进行相关的处理。

3.2 远程通信部分的设计与实现

网关与控制台的远程通信是通过局域网, Intranet, Internet 等多种方式来实现。通信所采用的协议根据网络类型的不同而不同。如果通过 Internet, 采用的协议是 TCP/IP 协议, 这时控制台与网关之间的通信可以采用连接方式和不连接方式进行通信。对于连接方式, 当有多个控制台连接到网关上时, 需要对每一个控制台建立一个连接。对于无连接方式, 则通过侦听数据报的方式来对请求进行响应。另外, 为了达到实时性的服务, 一般采用数据报方式服务; 对于重要的参数, 则采用确认方式的数据报服务。

下面是在 TCP/IP 协议基础上实现网关和控制台之间的数据报通信的方法:

为了捕获到在接受到消息及连接完成等情况下触发的事件, 需要用 CAsyncSocket 派生一个子类 CMySocket, 为子类的 OnAccept, OnReceive, OnSend, OnClose 消息添加处理函数, 添加代码以执行各种事件被触发时需要完成的任务。这可以通过 Class Wizard 来完成。

网关作为服务提供方, 需要至少两个 CMySocket 对象, 一个用于侦听连接请求, 另一个用于通信。这里将两个 CMySocket 对象命名为 m_ListenSocket 和 m_ConnectSocket。网关首先要在某一个端口进行侦听, 以探测是否有远程控制台的访问请求。

```
m_ListenSocket.Create(m_iPort); //m_iPort 变量是侦听的端口号
```

```
m_ListenSocket.Listen(); //调用父类的侦听函数
```

当接收到控制台的连接请求时, m_ListenSocket 对象的 OnAccept 事件被触发, 用一个新的 CMySocket 对象进行连接:

```
m_ListenSocket.Accept(m_ConnectSocket); // m_ConnectSocket 是用来建立连接的对象
```

之后网关接收数据并对数据进行处理, 如访问授权的验证, 命令的解释, 信息的反馈等。

对于远程控制台来说, 建立一个控制台程序(Console), 只需要一个 CMySocket 对象就可以实现通信了。

```
m_ConnectSocket.Create();
m_ConnectSocket.Connect(m_strGatewayURL, m_iPort);
//m_strGatewayURL 为网关的 IP 地址, m_iPort 为网关进行侦听的端口号
```

连接建立后, 就可以用网关和控制台上的连接对象 m_ConnectSocket 的 Send 和 Receive 方法来收发数据进行通信了。限于篇幅该方法的使用请参考 MSDN 里 CAsyncSocket 部分。

参考文献

- 1 涂时亮, 张有德著. 单片微机控制技术. 复旦大学出版社
- 2 张力, 吕振肃, 赵庆林. VC++ 下串行通信的编程方法. 计算机应用, 1999(3)
- 3 Jerry Anderson 著. 张之一, 史元春译. Visual C++ 5 ActiveX 编程指南. 清华大学出版社
- 4 TCP/IP Illustrated, Volume 2: The Implementation. Gray R. Wright and W. Richard Stevens
- 5 Microsoft Corp. Introduction to ActiveX Controls. MSDN

(上接第 129 页)

当 $N=M$ 时, 乘法次数:

$$(m-1)(6m+6) + (n-1)(6n+6)$$

用常规方法计算时推导可得乘法次数是:

$$3m^2 + 3(m-1)^2 + \dots + 3 + (2m-1) \cdot 3 + (2m-3) \cdot 5 + \dots + 3m + 3m^2 + 3n^2 + (2m+1)(2n+1) + 3n^2 + 3(n-1)^2 + \dots + 3 + (2n-1) \cdot 3 + (2n-3) \cdot 5 + \dots + 3(2n-1) + 3m^2 + 3n^2 + (2n+1)(2m+1)$$

明显看出快速双线性变换的乘法次数少于常规方法。

结束语 本方法简单、明了具有一定的实际应用价值, 其特点是: (1) 设计完成可以用计算机一次成功, 参数改变很容易, 方便地得到低通、高通、带通、带阻等各种滤波器, 既可以输出图形, 又可以输出数据, 图文并茂, 便于分析, 便于修改, 具有很大的通用性。(2) 在算法上多次应用迭代、子程序、原位运算等, 因此运算速度快, 占用内存单元少。(3) 程序编制由于用了汉字兼容的方法, 所有提示、输出全用汉字, 所以十分利于工程人员研究使用。