

利用 DS80C400 微控制器 构建拨号网络

随着技术的进步，现有的大型网络大大简化了通过微控制器监视和控制传感器/执行器的过程。现在，可以将信息通过网络传送到某个中心点进行分析和校正。对于此类应用，网络型微控制器 DS80C400 提供了一个现成的方案。除了配备更全面的外围设备外，DS80C400 的固化软件还实现了 TCP/IP 栈¹。微型因特网接口 (TINI[®]) 平台² 提供对 IP 网络的全面支持，另外还包含了一个 Java™ 虚拟机 (JVM)。DS80C400 已包含了以太网接口，TINI 运行环境 (TRE) 还可另外提供对采用点对点协议 (PPP) 的拨号网络的支持。PPP 一个非常引人注目的好处是，一条连接的两个端点可利用调制解调器，通过公用通信网络和 IP 软件基础结构进行通信。这就允许用户将嵌入式网络应用部署到那些以太网无法到达，但无所不在的电话交换网却可达的偏远分站 (图 1)。

网络型微控制器 DS80C400 提供了一个现成的方案，可以将信息通过网络传送到某个中心点进行分析和校正。

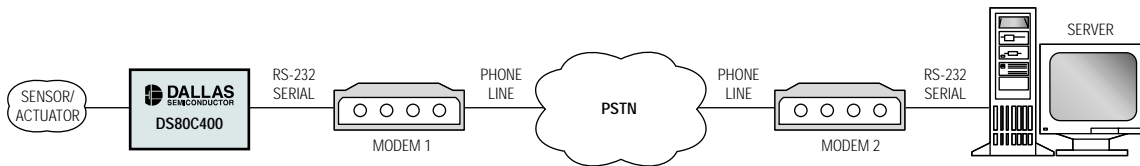


图 1. 一个运行 TINI 运行环境的远端 DS80C400 拨号连接服务器进行数据传递。

PPP 概述

PPP 是一种通用协议，支持多种物理媒体上的数据传送，包括 (但不限于) 串行、并行、以太网和蜂窝电话，例如通用分组无线业务 (GPRS) 设备。PPP 被广泛用于拨号网络应用，因为它仅要求很少的配置，并且建立起来非常容易。只要求物理媒体具有全双工能力。通信可以是同步或异步。

PPP 由三个主要部分构成：

- 1) 一种将多协议数据包封装到同一链路的方法。PPP 封装采用的是高级数据链路控制 (HDLC) 格式。如果带宽有限，有些封装域可以被压缩。
- 2) 一个用于建立连接、配置链路选项、检测错误和终止链路的链路控制协议 (LCP)。
- 3) 一组用于建立和配置相应网络层协议的网络控制协议 (NCP)。

内容列表

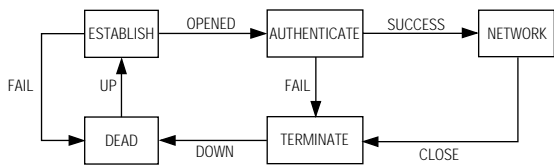
利用 DS80C400 微控制器构建拨号网络	1
基于 SRAM 的微控制器提供更优的安全性	8
采用 DS80C400 创建网络多媒体应用	12
利用 DS5240/DS5250 直接升级 DS5002	18

PPP支持多种物理媒体上的数据传输,包括串行、并行、以太网和蜂窝电话,例如通用分组无线业务(GPRS)等设备。

PPP 操作

因特网标准 RFC 1661 描述 PPP 操作作为一个状态机,随着点对点链路的配置、维持和终止,执行不同的进程。图 2 给出了一个简化的状态图,其中 PPP 被划分为五个不同阶段:等待,建立,认证,联网和终止。PPP 执行除认证之外的所有阶段。

链路等待阶段:链路操作的起始和结束阶段。物理层尚未准备好传送数据包。当物理层准备就绪时,产生一个 UP 事件,PPP 前进到链路建立阶段。



链路建立阶段:物理层就绪,通过交换 LCP 配置包,与对端磋商传送选项。在此阶段只配置那些独立于网络层协议的选项。一旦有配置 ACK 包被发送并接收,PPP 产生一个 OPENED 事件并前进到下一阶段。

认证阶段:用于对端认证的可选阶段。在有些链路中,例如拨号网络,常常希望链路在交换网络层协议包之前进行认证。为此,在链路建立阶段必须发送认证请求。

图2. 这个简化的流程图展示了 RFC 1661 中所描述的 PPP 实现。

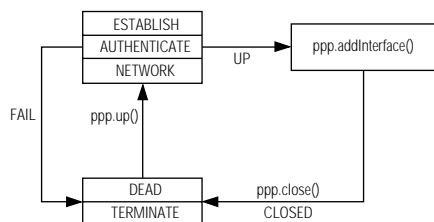
网络层协议阶段:在链路已经建立并成功通过认证之后,通过交换网络层所支持的特定 NCP 包对网络层协议进行配置。每个网络层协议具有一个特有的 NCP,并且需要进行单独磋商。

链路终止阶段:当 PPP 链路因下述原因被终止时产生一个 CLOSE 事件:载波丢失,认证失败,链路质量失败,或指令要求关闭链路。对端之间交换 LCP 终止包。通知网络层协议链路即将关闭,并采取适当动作。当接收到一个终止 ACK 或超时后物理层被关闭。然后产生一个 DOWN 事件,PPP 返回链路等待阶段。

TINI PPP

TINI 以 RFC 1661 为框架实现 PPP。PPP 通过某条串行链路,为 IP 数据包提供精确的数据传送服务。在本地网络栈中,PPP 位于 IP 模块的下方,串行端口驱动器的上方。为了降低编程的复杂度,PPP 进程被进一步简化了(图 3)。

在 `com.dalsemi.tininet.ppp` 包中,对于应用开发者,PPP 被解释为 Java 类。PPP 进程受事件驱动。`ppp.up()` 建立链路、认证并设置网络协议。支持口令认证协议(PAP)和质询-握手-认证协议(CHAP)³。一旦链路配置完毕,产生一个 UP 事件并添加 PPP 接口到网络栈中,以便网络业务可以被定向到该接口。`ppp.close()` 发出一个 CLOSE 事件,关闭链路并返回等待/终止状态。



例 1 所示片段来自 DS80C400 上实现的一个 PPPClient (访问 www.ibutton.com 并搜索 PPP 可找到最新实例)。PPP 对象生成后,PPPClient 被装载作为 PPP 对象的 PPPEventListener。设置 PPP 参数,并用一系列 `atCommand` 命令初始化链路。一旦链路建立,调用 `ppp.up()` 通知网络栈 PPP 已可用于网络业务。链路终止采用 `ppp.close()`。

图3. 这个 TINI PPP 流程图展示了 PPP 是如何在 TINI 上实现的。

例1. PPPClient实现

```
public class PPPClient extends Thread
    implements PPPEventListener, CommPortOwnershipListener{
    ...
    public void run(){
        ppp = new PPP();
        openSerialPort(portNumber);
        // Add this object as a PPP event listener
        ppp.addEventListener(this);
        // Set the local and remote IP address
        ppp.setLocalAddress(localAddress);
        ppp.setRemoteAddress(remoteAddress);
        // Set client peer type options
        // Set the ACCM to escape all octets
        ppp.setRemoteAccm(0x00000000);
        ppp.setLocalAccm(0x00000000);
        ppp.setAuthenticate(false, true);
        // Set username and password
        ppp.setUsername(username);
        ppp.setPassword(password);
        // Initialize modem
        for (int i = 0; i < dialSequence.length; ++i)
            atCommand(dialSequence[i]);
        // Set connected flag
        connected = true;
        // Issue up command to PPP FSM
        ppp.up(serialPort);
        // PPP connection is now established, we can now
        // communicate with remote host
        sendData();
        ppp.close();
        closeSerialPort();
    }
    ...
}
```

TINI运行环境提供的用户友好API将细节隐藏起来,使设计者集中精力于他们的设计,将PPP作为一个工具。甚至在传统电话网的情况下,用GPRS无线电话代替调制解调器,仍然可以运行该应用。

例2演示了如何处理一个PPPEvent。一旦链路已经就绪,并已收到UP事件,PPP被作为一个网络接口添加进来,然后就可传送IP包到该接口。无论何时收到一个CLOSE事件,网络栈随即卸掉PPP接口,终止任何通过PPP的网络活动。

例2. PPPEvent方法

```
/**
 * PPP event listener interface
 */
public void pppEvent(PPPEvent ev){

    switch (ev.getEventType()){
        case PPPEvent.UP:

            // PPP connection is up
            ppp.addInterface(interfaceName);
            interfaceActive = true;
            break;

        case PPPEvent.CLOSED:
            // PPP connection is closed
```

```

    if (interfaceActive){
        interfaceActive = false;
        ppp.removeInterface(interfaceName);
    }
    connected = false;
    break;
default:
    break;
}
}
}

```

远程湿度数据采集器举例

本文的其余部分，我们给出一个功能强大的网络应用，该应用充分利用了一种极为经济、外形小巧的计算机系统所提供的网络能力。本例采用了一种TINI参考设计，名为TINIIm400-030p, OEM Stamp+ Edition。该模块提供了一个低功耗、I/O丰富、低元件数的嵌入式控制器和数据采集模块。当与TRE相结合时，仅需极少的软件工作量便可组成一个健壮的网络型数据采集系统。该Stamp+模块包含闪存ROM、RAM、一个实时时钟、1-Wire®网络、并行I/O和异步串口等。由于设计目标是远程拨号应用，这个特定设计并没有使用以太网接口，尽管只需添加一个以太网PHY及相关的磁性元件即可实现。

本文给出了一个采集和记录数据的完整实例⁴，利用PPP管理的拨号网络通过PSTN(公用电话交换网)进行远程连接，为远端服务器提供数据。在拨号网络的支持下实现真正的远程数据采集。

*Application Note 702:
Using TINI Point-To-Point Protocol (PPP)*
演示了如何设置TINI使用PPP，通过串行链路传送IP包。请访问www.maxim-ic.com/appnoteindex。

系统概述

图4显示了利用模拟调制解调器构建拨号网络的装置。如果没有电话线或相似设备，硬件连线的串行到串行连接也可用于此测试装置。

图4中的试验配置包括下列设备：

- 一个PPP模块——运行DataLogger服务程序
- 一台Windows® 2000机器——运行DataLoggerServer软件
- 两个模拟调制解调器——一个安装于Win2K PC，另一个安装于PPP模块的串口
- 湿度感测电路——采集用于记录的湿度数据

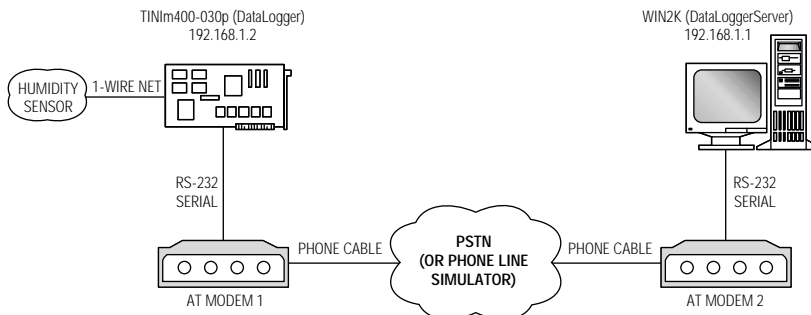


图4. 该远程数据采集系统利用普通调制解调器传送数据。

Stamp+ 模块上的DS80C400具有已安装好的TRE。TRE平台支持ASM、C和Java编程。嵌入式固件实现TCP/IP栈并为PPP协议提供框架。术语“TINI”，作为Stamp+模块远程数据采集功能的提供者，常常被和“Stamp+模块”交替使用。

PPP连接通过连接在仿真电话线两端的两个模拟调制解调器实现。如果有两条不同电话线，也可用公用电话网代之。要测试PPP接口，必须首先生成拨号网络连接。一旦启动连接，将发生下列一系列事件：

- 1) TINI 调制解调器拨号服务器的调制解调器。
- 2) 服务器调制解调器应答呼叫。
- 3) PPP 选项磋商开始。
- 4) TINI 发送认证信息给远端服务器。
- 5) 服务器安排一个 IP 地址给 TINI，并告知 TINI 服务器的地址。

软件和硬件概述

完整的源代码可从 ftp://ftp.dalsemi.com/pub/tini/reference_designs/TINIm400-030p/DataLogger.zip 下载。图5显示，湿度数据由传感器电路收集。本例采用的传感器为 DS1922H* 1-Wire 温度/湿度传感器，以 iButton® 形式封装，任何其他类型的 1-Wire 器件也可用于 DS80C400。

TINI 客户端软件

TINI DataLogger 范例演示了三个概念：1-Wire 网络，串行通信和 TCP/IP 网络。以下给出几个最重要的类的简要功能说明。

DataLogger 类

- 从配置文件(/etc/dataLogger.properties)取回参数
- 生成一个 HumidityLogger 的实例用于捕获采样
- 生成一个 PPPDaemon 的实例用于管理 PPP 连接
- 启动经由网络接口(例如以太网或 PPP)到远端服务器的出站连接

HumiditySensor 类

- 控制与 DS1922H 的通信并从中取回数据

HumidityLogger 类

- 初始化并管理湿度传感器
- 写采集数据到给服务器的输出流

PPPDaemon 类

- 拨号客户端
- 利用 PPP 接口建立 TCP/IP 连接
- 管理物理数据链路

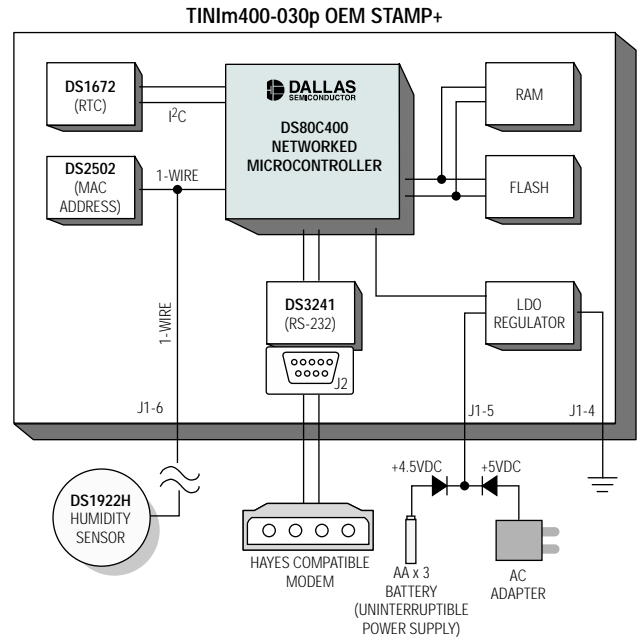


图5. TINIm400-030p 数据采集器利用一个传感器收集湿度数据。

关于 1-Wire 网络的详细描述，参见 *Application Note 148: Guidelines for Reliable 1-Wire Networks* 以及 *Technical Brief 1: 1-Wire Net Design Guide*，请访问 www.maxim-ic.com/appnoteindex。

* 未来产品——供货状况请联络厂方。

- 接收PPP事件通告
- 通知DataLogger 发生在物理数据链路上的错误

PPPSerialLink 类

- 实现PPPSerialLink 接口
- 允许PPPSerialLink 管理数据链路
- 配置串行口用于数据链路

PPPModemLink 类

- PPPSerialLink 的子类
- 管理调制解调器的通信
- 监视SerialPortEvent.CD (载波帧测)以检测调制解调器是否挂起

ModemCommand 类

- 控制与调制解调器的串行通信
- 等待预期响应超时时放弃DataLinkException

远程数据采集服务器

DataLoggerServer 是一个简单的GUI服务器应用程序，用于接受来自TINI的连接和下载其当前记录。

DataLoggerServer 类

- 显示采集数据
- 阻塞accept 以便等待PORT上的套接字连接
- 建立记录和绘制湿度与温度随时间变化的图表

运行本例

应用文件传统上是以太网接口，采用FTP协议传送到TINI文件系统的。由于Stamp+ 模块未包含以太网接口，在Slush和JavaKit中已添加了ymodem文件传送协议。Ymodem可以通过JavaKit 串行链路传送文件到TINI文件系统。更多关于Stamp+ 模块的详细信息，请参考 *Application Note 611 : Dial-Up Networking with the TINIm400 Stamp*。除了应用文件外，包含以下文本的文件DataLogger.tini, a /etc/.startup 也应传送到TINI文件系统。

```
#
# Starting DataLogger application from .startup file
#
setenv FTPServer disable
setenv TelnetServer disable
setenv SerialServer disable
#
```

```
initializeNetwork
#
java /DataLogger.tini
```

该启动文件禁止串行服务，允许数据采集应用访问串行端口。一旦应用和启动文件传送完毕，复位TINI便可处理新的启动文件并启动数据采集器应用。

TINI发送给DataLoggerServer的第一条信息是一个整数值，它告诉服务器即将到来的记录数量。读取该值后，服务器将所有记录环形排列，然后读取每个独立的采样。服务器显示每个登入的数据，并将这些信息记录到一个文件。DataLoggerServer用Java编写，运行时需要Java运行环境，运行Slush也要用到同样的运行环境。安装方法参见 www.java.sun.com。

运行DataLogger数分钟后(以便其获得若干采样)，DataLogger-Server开始运行。在本例中，每个采样均带有时间标记，且前后相隔一分钟。若DataLogger运行超过一小时，它便会填满采样记录表，产生60个(MAX_SAMPLE)数据采样。如果运行数天、数周甚至数月，它仍然保持MAX_SAMPLE个采样，但它们始终是最近一小时的读数。

结束语

在DS80C400上实现拨号网络连接简单直观。TINI运行环境提供的用户友好API将细节隐藏起来，使设计者集中精力于他们的设计，将PPP作为一个工具。甚至在传统电话网的情况下，用GPRS无线电话代替调制解调器，仍然可以运行该应用。

TINIm400 Stamp可被配置为启动或接收拨号连接。对于数据采集应用，中央服务器可周期性地拨通TINIm400 Stamp并取回数据。当发生本地故障时，Stamp模块可启动PPP拨号连接到中央服务器，将错误通知系统。本地故障由TINIm400检测，这样中央服务器可专注于数据的分析工作。

同任何嵌入式模块一样，硬件和算法依赖于特定的应用。DS80C400丰富的I/O资源和TINI运行环境的灵活性使设计者能够十分方便快捷地添加远程传感器/执行器到网络中。

TINI, 1-Wire和jButton是Dallas Semiconductor的注册商标。
Java是Sun Microsystems的一个商标。
Windows是Microsoft Corp.的一个注册商标。

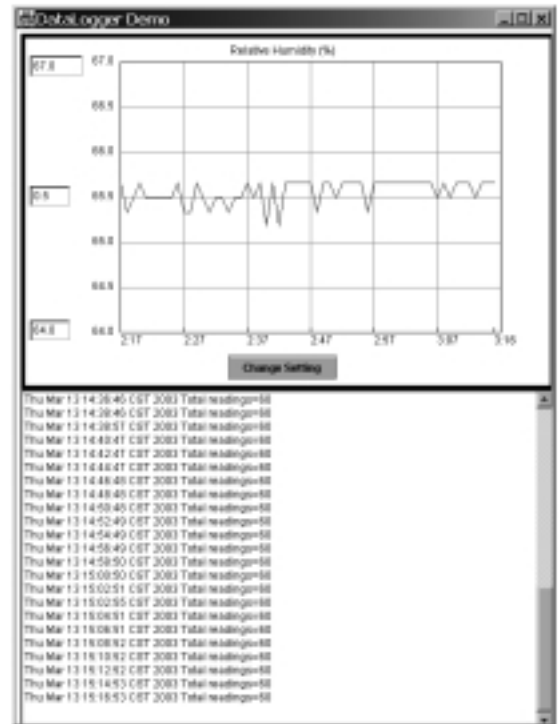


图6. DataLoggerServer工作时的PC屏采样显示。

脚注：

- 1 硅软件支持以太网上的IPv4/6。
- 2 Application Note 708 : Tiny InterNet Interfaces (TINI)
- 3 TRE固件版本1.1或更新。
- 4 本例摘自 *The TINI Specification and Developer's Guide* 之第7章，见 www.maxim-ic.com/TINIGuide。

全部相关的应用笔记列表参见
www.maxim-ic.com/appnoteindex。