

# uC/OS II 下协议栈 uIP 的移植与应用

刘春风<sup>1</sup>, 张代远<sup>1,2,3</sup>

- (1. 南京邮电大学 计算机学院, 江苏 南京 210003;
2. 江苏省无线传感网高技术研究重点实验室, 江苏 南京 210003;
3. 南京邮电大学 计算机技术研究所, 江苏 南京 210003)

**摘要:**随着互联网的发展,各种家用电器从空调到冰箱,都产生了接入 Internet 的需求。针对这一要求,文中把一个轻量级的 TCP/IP 协议栈 uIP 移植到基于三星的 S3C2410 处理器和 uC/OS II 操作系统的目标机上。移植步骤与方法如下:uIP 的配置,以太网接口芯片驱动程序的编程,S3C2410 定时器编程,uC/OS II 操作系统的移植,基于 uIP 的应用程序设计。最后将移植后的 uIP 应用到一个远程报警系统中,实现了远程报警功能。实验结果表明文中成功的在智能设备和 Internet 间建立了通信链路,实现了智能设备与 Internet 的互联。

**关键词:**TCP/IP; uIP; 移植; uC/OS II

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2012)09-0143-03

## Porting and Application of uIP in uC/OS II

LIU Chun-feng<sup>1</sup>, ZHANG Dai-yuan<sup>1,2,3</sup>

- (1. College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China;
2. Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing 210003, China;
3. Institute of Computer Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract:** At present, with the rapid development of the Internet technology, a variety of household appliances, from air conditioning to the refrigerator, produced the needs of access to the Internet. In response to this request, put a light weight TCP/IP protocol stack uIP porting to S3C2410 processor based on uC/OS II operating system on the target board, the main work is as follows: uIP configuration, ethernet interface chip driver programming, S3C2410 timer programming, uC/OS II porting, application design based on uIP. Finally, applied ported uIP to a remote alarm system, realized the remote alarm. Experimental results show that the link between smart devices and Internet communications is successfully established to achieve the interconnection of them.

**Key words:** TCP/IP; uIP; porting; uC/OS II

## 0 引言

在一个由 32 位嵌入式处理器构建的中、高端网络接入嵌入式系统中,通常会运行一个集成 TCP/IP<sup>[1-3]</sup> 协议栈的操作系统。但是对于系统资源有限、产品价格要求严格的嵌入式系统,由于它所具有的处理能力十分有限,通常不在它里面运行操作系统,这就要求系统开发者根据应用的要求选择合适的处理器并且构建自己的 TCP/IP 协议栈。TCP/IP 协议的透明性掩盖了

它实现的复杂性,从无到有构建一个协议栈是一项艰巨的任务,并且缺少有效的调试工具。uIP 协议栈是一个能够运行于 8 位到 32 位微处理器的轻量级的 TCP/IP 协议栈,它可以自由分发和应用用于商业或非商业目的。uIP 协议栈是使用 C 语言编写,便于移植,并且 uIP 协议栈编译后的代码大小对 RAM 的需求比其它一般的 TCP/IP 协议栈要小,这就使得它能够方便地应用到各种低端系统上。

## 1 系统分析与移植步骤

### 1.1 配置 uIP

移植 uIP 时,首先需要配置一些选项,目的是适应具体的硬件平台,这里的选项就是一些宏定义,包括: UIP\_CONF\_BYTE\_ORDER 数据存储方式(大端或者小端), UIP\_CONF\_BUFFER\_SIZE(uIP 缓冲区大小),主

收稿日期: 2012-02-02; 修回日期: 2012-05-12

基金项目: 江苏高校优势学科建设工程资助项目(yx002001)

作者简介: 刘春风(1988-),女,江苏徐州人,硕士研究生,研究方向为智能计算技术与应用;张代远,教授,硕士生导师,研究方向为智能计算理论、方法与应用,计算机体系结构,计算机在通信中的应用。

机 IP 地址,主机 MAC 地址,默认路由,子网掩码,uIP 一秒时间所需要中断数,基本数据类型的定义,UIP\_APPCALL 定义 uip\_tcp\_appstate\_t 定义<sup>[4]</sup>等。

### 1.2 以太网接口芯片编程

文中所用的以太网接口芯片是 RTL8019AS<sup>[5]</sup>。网卡驱动层与底层硬件密切联系,它为 uIP 提供服务,屏蔽了硬件实现的细节,系统中各层次关系如图 1 所示。对于文中的网卡驱动部分,只需实现三个函数:char rtl8019as\_init(void) 网卡初始化函数;void rtl8019as\_send(void) 网卡发送数据包函数;unsigned int rtl8019as\_read(void) 网卡接收数据包函数。编程知识可参考文献<sup>[6~8]</sup>。

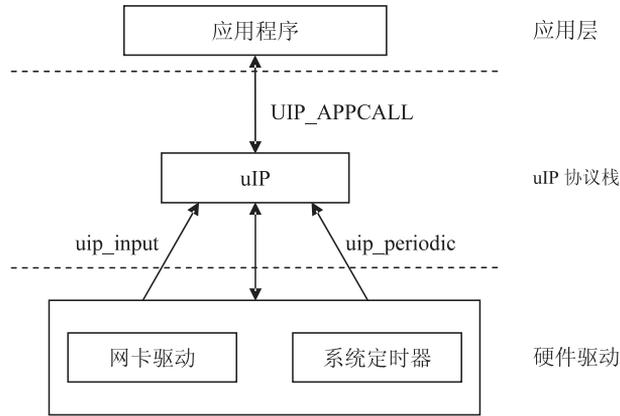


图 1 系统中各层次之间的关系图

初始化:RTL8019 的初始化主要设置接收和发送缓存,设置读页指针和写页指针,配置接收和发送模式,同时需要设置本机 MAC 地址。由于使用的是 uIP 网络协议栈,因此网卡的 MAC 需要与协议层封装的 MAC 统一,因此使用预先定义好的宏定义 UIP\_ETHADDR0 -UIP\_ETHADDR5。当然也可以从 EEPROM 中读取,可实现动态修改 MAC 的功能。

发送数据包:对 uIP 网络协议栈来说,需要发送的数据已经存放在全局内存 uip\_buf 里,数据包长度大于 54 字节(TCP+ETH 头部)时,使用 uip\_appdata 指向应用层数据,而需要发送数据的长度使用 uip\_len 记录。因此,发送数据的步骤:

(1)启动远程写,将 uip\_buf 中的数据写到 RAM BUFFER 的发送缓冲区中。

(2)将发送的首页赋给 TPSR,长度(小于 60 时算 60)赋给 TBCR1, TBCR0。启动发送。

当然,放在 uip\_buf 的前 14 个字节按照以太网帧的 DA,SA,TYPE 排列好的,这样可以发送到对应 MAC 地址的网卡中。

接收数据包:uIP 的数据接收后放在全局内存 uip\_buf 中,接收的数据长度放在 uip\_len 中,因此,发现一个完好的以太网帧后,将前 4 个字节去掉后依次存放

在 uip\_buf 中。RTL8019 收包采用查询模式,当发现 RAM BUFFER 接收缓冲区中有未处理的数据时,先读取 4 字节,计算出数据的长度和下一帧的页地址,然后启动远程读操作,将数据存放在 uip\_buf 中。当然,还可以根据网络的实际情况增加一些传输中产生错误(如接收缓存满)的处理代码。

以太网接口芯片在总线中的基地址必须配置正确,这是访问以太网卡中其他寄存器的基础。以太网接口芯片接收,发送数据包的流程如图 2 和图 3 所示。

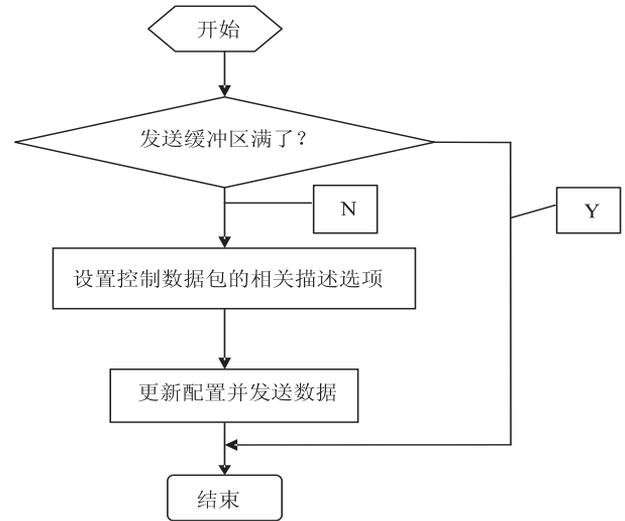


图 2 网卡发送数据流程图

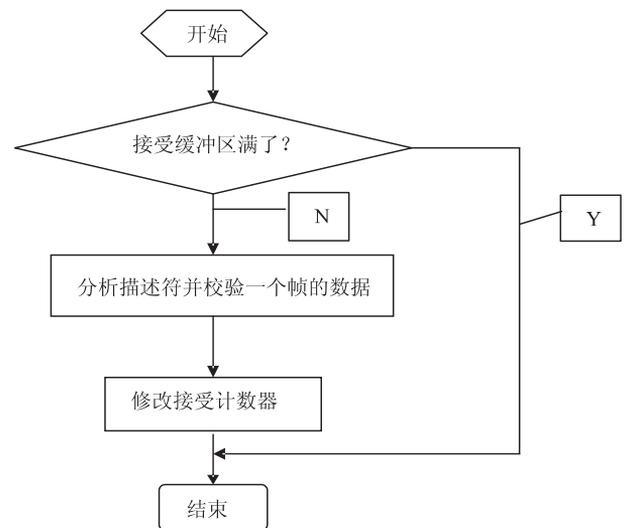


图 3 网卡接收数据流程图

### 1.3 S3C2410 定时器编程

S3C2410 共有 5 个定时器<sup>[9]</sup>,其中 0、1、2、3 有 PWM 功能,且有一个输出引脚,可以通过定时器来控制引脚周期性的高、低电平变化;定时器 4 没有输出引脚。

定时部件的时钟源为 PCLK,首先通过两个 8 位预分频器降低频率;定时器 0、1 共用第一个预分频器;定时器 2、3、4 共用第二个预分频器;预分频器的输出将进入第二级分频器,输出 5 种频率的时钟:2 分频、4 分

频、8 分频、16 分频、外部时钟 TCLOCK0/TCLK1;每个定时器的工作时钟都可以在这 5 种频率中选择;系统中 PCLK 的值 50.7MHz, 预分频值为 120, 二级分频值为 16, 这样, 定时器定时 1 秒需要计数的值为  $50700000/120/16 = 26406$ 。本系统中设置定时器 3 的初值为  $26406/100$ , 定时即为  $1/100$  秒, 所以 CLOCK\_CONF\_SECOND 的值即为 100。

#### 1.4 移植 uC/OS II

整个 uC/OS II 移植实现中, 只需要提供一个汇编语言文件, 提供几个必须由汇编才能实现的函数, 具体可参考文献[10, 11]。所需要实现的函数有:

##### 1) OSStartHighRdy()。

操作系统启动之后负责启动优先级最高的进程, 该函数只会被调用一次。该函数的实现方法是从当前就绪任务控制块中获取优先级最高的控制块指针 (addr\_OSTCBHighRdy), 通过该指针可以获得该任务的堆栈指针 SP, 通过该栈指针依次弹出状态寄存器的值, 通用寄存器的值以及 PC, LR 的值, 从而改变 CPU 现场, 使程序从弹出的 PC 值处开始执行。

##### 2) OSCtxSw()。

任务切换函数, 主要工作是保存当前任务的 CPU 现场, 然后从当前优先级最高的任务控制块中取得任务堆栈指针, 恢复 CPU 现场, 并执行。该函数的实现方法是通过当前任务控制块指针和就绪队列中最高优先级的任务控制块指针比较两个任务优先级, 如果就绪队列中任务的优先级高, 那么就切换执行高优先级任务, 切换方法与函数一的实现类似。

##### 3) OSIntExit()。

该函数是中断级的任务切换函数, 在时钟中断 ISR 中发现有高优先级任务在等待时, 需要在中断退出后不返回被中断的任务, 而是直接调度就绪的高优先级任务执行。其目的在于能够尽快让高优先级的任务得到响应, 保证系统的实时性能。

##### 4) OSTickISR()。

该函数是时钟中断处理函数, 主要任务是负责处理时钟中断, 调用系统实现的 OSTimeTick 函数, 如果有等待时钟信号的高优先级任务, 则需要在中断级别上调度其执行。另外两个相关函数是 OSIntEnter() 和 OSIntExit(), 都需要在 ISR 中执行。

#### 1.5 uIP 应用

有了先前的准备工作, 现在把各个模块整合起来, uIP 模块、网卡驱动模块、定时模块、uC/OS II 模块都已经成功移植进来。现在利用 uC/OS II 创建两个任务, 一个是网卡轮询任务, 另一个是应用程序任务, 当网卡接收到数据并读进缓存后就给 uip\_len 赋值, 值就为读进数据的长度。并向应用程序任务发出信号, 应

用程序接收到信号后, 就读写缓存中的数据, 并处理数据, 如果应用程序有数据要发送就向 uip\_buf[] 缓存中写入数据, 并给 uip\_len 赋值, 当 uIP 处理时, 如果发现 uip\_len 的值大于 0, 说明应用程序有数据要发送, 就调用网卡驱动发送函数进行数据发送。

系统刚启动时, 应用程序任务处于休眠状态。当网卡驱动程序任务接收到数据时, 就唤醒应用程序, 应用程序任务收到信号后就从 uip\_buf 中读取数据, 如果应用程序没有数据发送就直接返回, 如果有数据要发送就将数据存放到 uip\_appdata 所指向的内存单元中, 再调用 rtl8019as\_send() 函数发送数据。与此同时, 应用程序需每隔 10 秒对 uIP 的 ARP 表项进行更新, 同时每隔 0.5 秒对每个 TCP 连接进行轮询, 看是否有需要发送的数据, 保证对每个连接的实时响应。系统应用结构如图 4 所示。

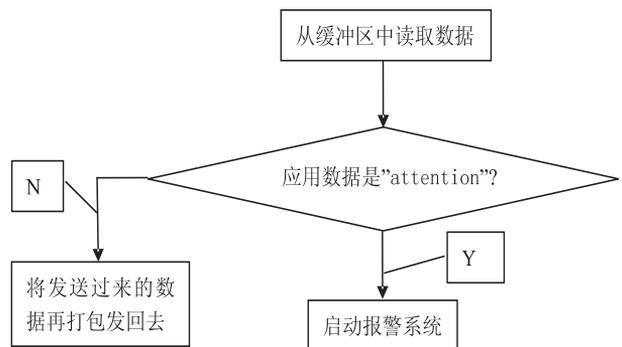


图 4 报警系统流程图

## 2 主要成果

(1) 将 uIP 顺利地移植到 uC/OS II 中, 使得互联网上的嵌入式设备能够与 PC 机正常通信;

(2) 将该成果应用到一个报警系统中, 只要远程的 PC 机运用 TCP 连接向目标设备发送“attention”字符串, 目标设备即可启动报警系统, 实现远程报警。

## 3 设计的创新点

通过使用 PC 机和目标板进行通信测试, 证明该系统实现了预先的设计要求, 并且有以下几个创新点:

(1) 将一个小型的 TCP/IP 协议栈 uIP 移植到了嵌入式设备上, 从而该设备可以跟互联网上的 PC 机进行通信;

(2) 本系统代码编译后只有几百 KB, 占用极少的内存资源;

(3) 该协议栈可配置性强, 适应不同资源环境 and 应用<sup>[12]</sup> 场合;

(4) 可以应用于多种需要接入网络的工控系统中。

$$\mu_{ex}(x,y) = \begin{cases} 1 & D(x,y) = \beta_{\max} \\ \frac{\beta_{\min}(x,y)}{\beta_{\max}} & 0 < D(x,y) \\ 0 & \text{其他情况} \end{cases} \quad (10)$$

文中是在文献[12]研究的基础上对复杂空间模糊面实体拓扑划分模型的改进,并给出相应的模糊隶属函数,在理论上,运用严格的数学方法,进行了详细的定义和形式化描述,该模型适用于不同复杂程度的模糊地理实体,有利于读者对空间模糊实体的划分有更加清晰的理解。

#### 4 结束语

文中在已有的关于对 GIS 简单要素(如点和面)的模糊形式化描述的基础上,进行了详细的分析和总结。

针对现有的关于对模糊点和模糊面拓扑划分的描述部分存在不足,文中在形式化描述和数学公式表达方面进行更详细、准确的描述,最后完成了从简单实体的模糊拓扑划分到描述复杂的地理实体的拓扑划分。

未来将进一步对在某个区域内存在多个不同地理特征的更加复杂的地理现象进行研究,并通过试验进行模糊空间划分。

#### 参考文献:

[1] Burrough P A, McDonnell R A. Principles of Geographical In-

(上接第 145 页)

#### 4 结束语

文中研究并实现了 uIP 协议栈在 uC/OS 下的移植与应用,可以实现 Internet 上的智能设备跟网络中所有的 PC 机进行通信,可以实现实时远程工业控制,而不需要人为的手动干预,控制者不需要在现场就可实现控制设备,大大减少了出差的次数,是未来嵌入式设备发展的一个趋势。

#### 参考文献:

- [1] 谢希仁. 计算机网络[M]. 第5版. 北京:电子工业出版社, 2008:63-220.
- [2] 沈辉. 计算机网络工程与实训[M]. 北京:清华大学出版社, 2002:40-125.
- [3] Comer D E. 计算机网络与因特网[M]. 第5版. 北京:机械工业出版社, 2009:35-170.
- [4] Dunkels A. The uIP 1.0 Reference Manual[D]. Swedish: Swedish Institute of Computer Science, 2006.

formation Systems[M]. Oxford:Oxford University Press, 1998.

- [2] Molenaar M. An introduction to the theory of spatial object modeling[M]. London: Taylor Fancis, 1998.
- [3] 刘文宝. GIS 空间数据的不确定性理论[D]. 武汉:武汉测绘科技大学, 1995.
- [4] 刘文宝, 邓敏. GIS 图上地理区域空间不确定性的分析[J]. 遥感学报, 2002, 6(1):45-49.
- [5] 邓敏, 李成名, 林宗坚. GIS 中模糊区域的形式化描述方法探讨[J]. 测绘科学, 2002, 27(1):39-41.
- [6] 唐新明, 方裕. 模糊区域拓扑关系模型[J]. 地理与地理信息科学, 2003, 19(2):1-10.
- [7] 刘文宝, 邓敏. 矢量 GIS 中模糊地理边界的分析[J]. 山东科技大学学报(自然科学版), 2000, 19(1):28-32.
- [8] 郭庆胜, 杜晓初. 不确定性线状目标之间拓扑关系的描述与判别[J]. 武汉大学学报·信息科学版, 2004, 29(9):827-831.
- [9] 李大军, 刘波, 程朋根, 等. 模糊空间对象拓扑关系的 Rough 描述[J]. 测绘学报, 2007, 36(1):72-77.
- [10] Egenhofer M J, Franzosa R. Point-set topological spatial relations[J]. International Journal of Geographical Information Systems, 1991, 5(2):161-174.
- [11] 邓敏, 刘文宝. GIS 面目标拓扑关系的形式化模型[J]. 测绘学报, 2005, 34(1):85-89.
- [12] 杜世宏. 空间关系模糊描述及组合推理的理论和研究方法研究[D]. 北京:中国科学院, 2004.
- [13] 陈学工, 张驰伟, 张文艺, 等. 度量参数与空间关系描述的研究[J]. 计算机技术与发展, 2007, 17(5):187-190.
- [5] Realtek Semi-conductor Co Ltd. Realtek Full-Duplex Ethernet Controller with Plug and Play Function Specification[S]. 2001.
- [6] C 程序设计[M]. 第4版. 北京:清华大学出版社, 2010:45-268.
- [7] 周作厚. C 程序设计[M]. 北京:中国铁道出版社, 2003:24-150.
- [8] 常玉龙. Turbo C 2.0 实用大全[M]. 北京:北京航空航天大学出版社, 1994:18-147.
- [9] S3C2410 中文数据手册 (Version 1.0) [M]. 杭州:杭州立泰宇电子有限公司, 2008.
- [10] 邵贝贝. UC/OS 一源码公开的实时嵌入式操作系统[M]. 北京:中国电力出版社, 2001.
- [11] 陈是知. uC/OS-II 内核分析、移植与驱动程序开发[M]. 北京:人民邮电出版社, 2007.
- [12] 伊文斌, 周贤娟, 鄢化彪, 等. uIP TCP/IP 协议分析及其在嵌入式系统中的应用[J]. 计算机技术与发展, 2007, 17(9):240-243.