

## 基于 CAN 总线技术的嵌入式网关设计

The Embedding Gateway Design based on CAN Bus Technology

(1.安阳工学院 2.上海交通大学) 钟家民<sup>1</sup> 王国才<sup>2</sup>  
ZHONG Jia-min WANG Guo-cai

摘要: 本文主要从软硬件方面描述了一个嵌入式网关的具体设计。硬件方面系统使用嵌入式微控制器作为处理器,不仅要能执行传统的控制功能,还要能执行与连接因特网相关的功能;软件设计分为三个模块:CAN 总线设备接口通信模块、透明网关协议转换模块和以太网层应用模块。

关键词: 嵌入式网关; CAN 总线技术; 微处理器  
中图分类号: TP368.1 文献标识码: A

**Abstract:** A new embedded gateway design is described in detail based on the software design and hardware design. For the hardware design, it adopts embedding micro-processor as MCU, which not only executes the traditional function, but also connect to the internet; For the software design, it's composed of three modules: CAN bus equipment interface communication module, purify gateway protocol transform module and Ethernet application module.

**Key words:** Embedding Gateway; CAN Bus Technology; MCU

### 1 引言

作为当今最具发展前途的网络技术,自 20 世纪 70 年代末出现以来,以超出所有人预期的速度迅猛发展,现在已渗透进科学技术的方方面面。在测控系统应用网络技术将极大提高系统的性能,计算机测控技术与网络技术的结合已是大势所趋。如今,互联网应用正在转到以嵌入式设备为中心。IA (Internet Appliance)概念现在甚为流行,这表明互联网应用进入嵌入式互联网的时代已经来临。本文设计了一个嵌入式网关用于 Ethernet 之间的互连。

### 2 系统硬件分析与设计

系统使用嵌入式微控制器作为处理器,以太网驱动芯片 RTL8019AS 经耦合隔离滤波器 HR61101G 和 RJ45 接口接入以太网,可以将采用 CAN 总线协议通信的设备连到以太网上。CAN 总线控制器采用 SJA1000 芯片,CAN 收发器则采用了 PCA82C250。

为适应上网需求,嵌入式微控制器不仅要能执行传统的控制功能,还要能执行与连接因特网相关的功能,而 SX52 芯片能比较容易地实现 TCP/IP 协议,所以我们选择它作为微控制器。实现嵌入式设备接入 Internet,从原则上讲最关键的就是要实现 TCP/IP 协议,还有一个关键的问题就是传输信息媒质的选择。嵌入式系统应用最广泛的网络驱动芯片就是 NE2000 兼容系列网络芯片,它具有接口方便、驱动简单、占用资源少等优点,特别适合嵌入式系统。我们选用 RTL8019AS 芯片。

#### 2.1 CAN 总线网络设备接口设计

CAN 总线网络设备接口设计较网关设计简单。它是在完成设备功能的基础上加入一个 CAN 通信控制器接口芯片,实现与 CAN 总线网络的连接。

#### 2.2 网关设计

嵌入式透明网关设计是整个系统的核心。它由 CAN 控制器协议转换模块和以太网控制器协议转换模块两部分组成。网关硬件中微处理器起核心作用。

数据的流向为: 请求和控制信息从局域网中来,通过 RJ45 送到 RTL8019AS,RTL8019AS 负责将以以太网帧的首部和尾部信息剥离,将处理后的数据包送入 TCP/IP 协议栈,由协议栈对数据包进行解析,得到原始的请求和控制信息。请求和控制信息再经过 SJA1000 进行 CAN 协议格式的数据封装,再和现场的 CAN 总线设备进行通信。请求和控制的回复领先局域网的过程与上面正好相反。如图 1。

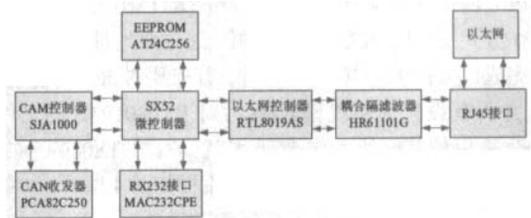


图 1 硬件结构框图

图中 AT24C64 为 8K EEPROM 主要是用来保存嵌入式透明 SX-52 网关的配置信息,如网关 IP 地址、MAC 地址和 SJA1000 的网网络标示符、网络掩码 MAR 和总线定时(BTR0, BTR1)等,这样,我们就能够灵活方便地修改网关参数,适应不同环境,同时也考虑到以后的扩展。硬件电路图如图 2 所示:

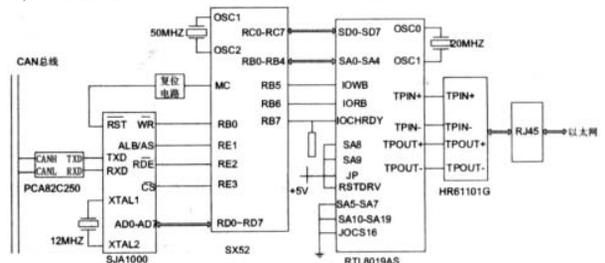


图 2 硬件电路图

### 3 系统软件设计

整个系统的软件设计可以分为三个部分:CAN 总线设备接口通信模块、透明网关协议转换模块和以太网层应用模块。

#### 3.1 协议转换模型

透明网关协议转换模型的设计整体思路为:当以太网应用层有数据要发送到 CAN 节点时,首先,数据发送到透明网关由以太网控制器协议转换模块从传输层数据报文中解析出完整的 CAN 协议数据包,存放在数据缓冲区 A,再通知总调度模块,由它调用 CAN 控制器协议模块将它发送到 CAN 总线上。反过来,当 CAN 设备有数据要发送到用户层时,首先,数据发送到透明网关由 CAN 控制器协议模块将完整的 CAN 协议数据包存放在数据缓冲区 B,再通知总调度模块,由它调用以太网控制器协议转换模块将完整的 CAN 协议数据包作为应用层数据封装起来,再发送到以太网的应用层。其模型结构图如图 3 所示

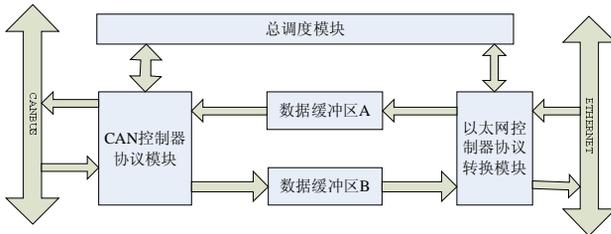


图 3 透明网关协议转换模型结构图

#### 3.2 CAN 控制器协议驱动模块

CAN 控制器协议转换模块程序主要由 SJA1000 的寄存器读程序 CANRead(),写程序 CANWrite(),初始化程序 CANInit(),总线状态查询 CANStatus()和发送、接收程序 txdsb(),rxdsb()组成。之所以要编写单独的 SJA1000 的寄存器读、写子程序,这是由 SX52 芯片只有 I/O 端口的特点决定的。

##### 3.2.1 以太网控制器协议驱动模块

要将系统接入以太网,就要对 RTL801A9S 进行编程,完成以太网帧的数据收发,相当于实现 PC 机中网卡的驱动程序功能。以太网控制器驱动程序用于设置 RTL8019AS 的工作状态和工作方式,分配收发数据的缓冲区,通过对地址及数据口的读写来完成以太网帧的接收与发送。首先要对 RTL8019AS 进行复位,并将其设置为跳线模式,然后对 RTL8019AS 的工作参数进行设置,以使其开始工作;接下来就读写 RTL8019AS 的 RAM 以完成数据包的接收和发送。程序流程如图 4 所示。

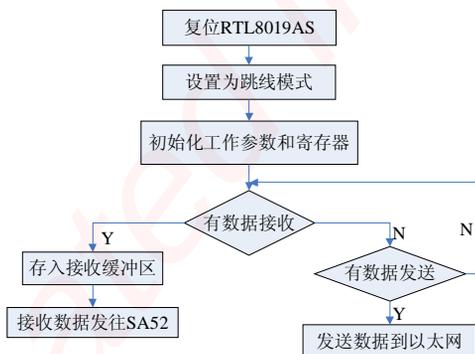


图 4 RTL8019AS 工作流程

以太网控制器 RTL8019AS 工作在网络中的数据链路层,对于它的应用主要有 3 个基本函数,即 RTL8019AS 初始化,状态检测函数、数据帧发送和接收函数。其中状态检测函数用来监

测是否有数据帧到达以及总线占用状态。RTL8019AS 有两个 DMA 通道,一个是本地 DMA,用做缓冲 RAM 与 FIFO 之间的数据交换,具有较高优先级,远程 DMA 用作外部存储器与 RTL8019AS 内部缓冲 RMA 之间的数据交换。SX52 通过远程 DMA 将一个数据帧放入 RTC8019AS 内部的缓冲区 RAM 中后,利用 I/O 方式将发送起始地址送到 TPRS,帧长度送到 TBRC,然后将命令寄存器 RC 中 TXR 置位开始一个数据帧的发送。当网络上有数据到达 RTL8019AS 时,它的 CURR 将不等于 BNRY,如果采用中断方式,则 RTL8019AS 将会自动通知 MCU,然后可采用包发送将缓冲环中的数据取出,这里我们采用查询方式来处理。

##### 3.2.2 CAN-Ethernet 协议转换模块

主程序工作流程图如图 5 所示,单片机首先初始化网络设备。网卡 IP 地址和物理地址存于单片机的 EEPROM 中,单片机复位后首先读取这些数据以初始化网卡。

网卡初始化完成以后,当有数据从 RJ45 过来时,单片机对数据包进行分析,如果是 ARP(物理地址解析协议)数据包,则程序转入 ARP 处理程序。如果是 IP(网间协议)数据包且传输层使用 UDP(用户数据报协议),端口正确,则认为数据报正确,数据解包后,送入数据缓冲区 A,最后把数据部分通过 CAN 接口输出。反之,如果单片机从 CAN 接口收到数据,则将数据按照 UDP 协议格式打包,送入数据缓冲区 B,由 RTL8019AS 将数据输出到局域网中。

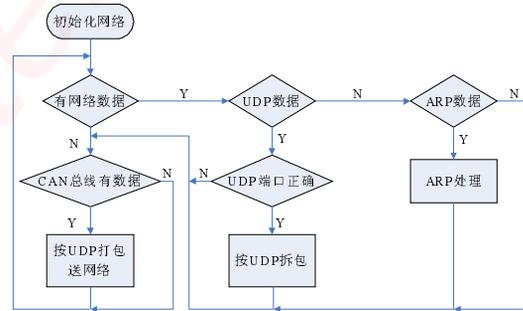


图 5 主程序工作流程图

这里,我们在通信传输层采用 UDP 协议是考虑到 CAN 协议数据报为短帧形式(每个数据帧最多为 8 个字节),如果采用 TCP 传输协议,我们要传输 8 个字节 CAN 协议数据,就要先通过 3 次握手建立连接,再传输数据,之后还要通过握手释放连接,这样传输效率对有限的网络资源来说无疑是一种浪费。而 UDP 是无连接的传输,这样可以提高网络传输效率,同时,也减轻网关的处理任务。当然,UDP 传输协议是不可靠的,对于控制网络来说,这是不允许的。

以太网控制器协议转换模块主要由以太网卡驱动、ARP、ICMP、UDP、TCP 协议的若干个 API 函数组成。

本文作者创新点:

本文设计了一个嵌入式多协议 CAN 网关,选用 8 位微控制器作为核心控件,通过控制以太网控制芯片和 CAN 总线控制芯片与 CAN 总线相连。

本项目经济效益为 125 万元。

#### 参考文献

- [1]Ubicom Corporation. IP2022 DataSheet. 2002。
- [2]李丹美. 网络处理器 IP2022 的原理及应用[J]. 国外电子元件,2002,5:31-34。

调用 rssiRsp()把收集到的 blast 信息的 RSSI 求平均数,然后加上自己的位置信息发送给 BlindNode。RefNode 的无线定位功能伪代码如下:

```
RefNode_ProcessEvent( byte task_id, uint16 events )
{
    if ( 事件为系统消息事件 ) //跨任务之间的消息传递都是系统事件 ZigBee 协议栈的元语都属于系统事件
        从消息队列中取消息
        switch ( 消息所包含事件 )
        {
            case AF 层有数据到来
                processMSGCmd( MSGpkt )
                {
                    switch ( pkt->clusterId ){
                        case LOCATION_XY_RSSI_REQUEST:
                            向 BlindNode 发送 XY-RSSI Response
                            消息
                            case LOCATION_RSSI_BLAST:
                                收集收集到的 blast 消息
                                .....
                            }
                        .....
                    }
                }
            .....
        }
    }
    .....
}
```

## 结束语

以 CC2430/CC2431 为硬件核心,基于 ZigBee 技术的井下综合人员管理系统由于使用了 Z-stack 和 OSAL,拥有了良好的协议栈性能和操作系统出色的多任务协作能力,由此从设计上对目前的井下人员定位系统和井下环境监测系统进行了融合,并扩展了良好的用户界面,使得系统能够完成人员定位,环境监测,报警,生产指令通过液晶屏显示等一些功能,并且借助 CC2431 的 RSSI 定位引擎,定位精度有了很大提高,在实验中达到了 1.5 米左右的定位精度。通过多次实验,系统基本符合设计目标,目前系统已经完成了无线传感网络部分的设计,上位机程序的设计正在进行之中,计划在上位机程序中使用 GIS 技术,给用户以更好的界面,动态显示人员在井下的行踪。

创新点:系统使用了 OSAL 作为操作系统,因此实现了复杂的应用程序,在一个模块上实现了人员定位,环境监测,报警等多项功能,由于使用了带定位引擎的 CC2431,因此定位精度与之前的一些定位系统相比有了显著的提高;由于使用了 ZigBee 技术,实现了双向通信,系统不仅能够实现定位和环境监测,还能传递上位机的信息。

项目经济效益:20 万左右

## 参考文献

[1]孙利民,李建中,陈渝,朱红松. 无线传感器网络. 北京:清华大学出版社, 2005  
 [2]IEEE Computer Society.IEEE Std 802.15.4?-2003 ,May 2003  
 [3]ZigBee Alliance. ZigBee Specification [ EB/ OL ] . http : / / www . zigbee . org . ,2005205210.  
 [4]Texas Instruments .Z-Stack Developer's Guide \_F8W-2006-

0022 Version 1.0  
 [5]Texas Instruments .Z-Stack HAL Porting Guide F8W-2006-0024 Version 1.0  
 [6]Texas Instruments . Z-Stack OS Abstraction Layer API F8W-2003-0002 Version 1.4  
 [7]俞海.基于 ZigBee 技术的矿井人员定位系统研究与设计.合肥工业大学.硕士学位论文.2007-5  
 [8]葛晓宇,王庆辉,魏立峰.ZigBee 技术及其在矿山中的应用[J]. 微计算机信息.2007,5-2:44-45  
 作者简介:薛毅飞(1983-),男,河南洛阳人,硕士研究生,主要研究方向:嵌入式操作系统、无线传感网络、嵌入式 linux 应用。  
**Biography:** XUE Yi-fei (1983-), male, postgraduate student, Research Area: Application of embedded linux, wireless sensor networks  
 (221008 江苏徐州 中国矿业大学 计算机科学与技术学院) 薛毅飞 马海波 姜薇 王潜平  
 (China University of Mining and Technology, Xuzhou 221008 China) XUE Yi-fei MA Hai-bo JIANG Wei WANG Qian-ping  
 通讯地址:(221116 江苏徐州 中国矿业大学南湖校区 计算机科学与技术学院计硕 06 班) 薛毅飞

## (上接第 60 页)

[3]SN75LBC184 具有瞬变电压抑制功能的差分收发器 [Z]. www.icbase.com。  
 [4]74LVX4245 DataSheet[Z]. www.fairchildsemi.com。  
 [5]FB2022 DataSheet[Z]. www.bothhand.com。  
 [6]Uvicom Corporation. IP2022 Programmer's Reference Manual, 2002。  
 [7]陈静. 一种新型网络处理器—IP2022[J]. 现代电子技术, 2003,2:29-33。  
 [8]吴振纲,陈虎.PLC 的人机接口与编程[J]. 微计算机信息, 2005 8-1:21-23。  
 作者简介:钟家民(1973.08-),男,汉族,河南淮阳人,安阳工学院讲师,硕士,主要从事教学软件开发、软件工程技术;王国才(1971-),男,安徽天长人,上海交通大学副教授,工学博士,主要从

# 嵌入式资源免费下载

## 总线协议:

1. [基于 PCIe 驱动程序的数据传输卡 DMA 传输](#)
2. [基于 PCIe 总线协议的设备驱动开发](#)
3. [CANopen 协议介绍](#)
4. [基于 PXI 总线 RS422 数据通信卡 WDM 驱动程序设计](#)
5. [FPGA 实现 PCIe 总线 DMA 设计](#)
6. [PCI Express 协议实现与验证](#)
7. [VPX 总线技术及其实现](#)
8. [基于 Xilinx FPGA 的 PCIE 接口实现](#)
9. [基于 PCI 总线的 GPS 授时卡设计](#)
10. [基于 CPCI 标准的 6U 信号处理平台的设计](#)
11. [USB30 电路保护](#)
12. [USB30 协议分析与框架设计](#)
13. [USB 30 中的 CRC 校验原理及实现](#)
14. [基于 CPLD 的 UART 设计](#)
15. [IPMI 在 VPX 系统中的应用与设计](#)
16. [基于 CPCI 总线的 PMC 载板设计](#)
17. [基于 VPX 总线的工件台运动控制系统研究与开发](#)
18. [PCI Express 流控机制的研究与实现](#)
19. [UART16C554 的设计](#)
20. [基于 VPX 的高性能计算机设计](#)

## VxWorks:

1. [基于 VxWorks 的多任务程序设计](#)
2. [基于 VxWorks 的数据采集存储装置设计](#)
3. [Flash 文件系统分析及其在 VxWorks 中的实现](#)
4. [VxWorks 多任务编程中的异常研究](#)
5. [VxWorks 应用技巧两例](#)
6. [一种基于 VxWorks 的飞行仿真实时管理系统](#)
7. [在 VxWorks 系统中使用 TrueType 字库](#)
8. [基于 FreeType 的 VxWorks 中文显示方案](#)

9. [基于 Tilcon 的 VxWorks 简单动画开发](#)
10. [基于 Tilcon 的某武器显控系统界面设计](#)
11. [基于 Tilcon 的综合导航信息处理装置界面设计](#)
12. [VxWorks 的内存配置和管理](#)
13. [基于 VxWorks 系统的 PCI 配置与应用](#)
14. [基于 MPC8270 的 VxWorks BSP 的移植](#)
15. [Bootrom 功能改进经验谈](#)
16. [基于 VxWorks 嵌入式系统的中文平台研究与实现](#)
17. [VxBus 的 A429 接口驱动](#)
18. [基于 VxBus 和 MPC8569E 千兆网驱动开发和实现](#)
19. [一种基于 vxBus 的 PPC 与 FPGA 高速互联的驱动设计方法](#)
20. [基于 VxBus 的设备驱动开发](#)
21. [基于 VxBus 的驱动程序架构分析](#)

## Linux:

1. [Linux 程序设计第三版及源代码](#)
2. [NAND FLASH 文件系统的设计与实现](#)
3. [多通道串行通信设备的 Linux 驱动程序实现](#)
4. [Zsh 开发指南-数组](#)
5. [常用 GDB 命令中文速览](#)
6. [嵌入式 C 进阶之道](#)
7. [Linux 串口编程实例](#)
8. [基于 Yocto Project 的嵌入式应用设计](#)
9. [Android 应用的反编译](#)
10. [基于 Android 行为的加密应用系统研究](#)
11. [嵌入式 Linux 系统移植步步通](#)
12. [嵌入式 C++语言精华文章集锦](#)
13. [基于 Linux 的高性能服务器端的设计与研究](#)
14. [S3C6410 移植 Android 内核](#)
15. [Android 开发指南中文版](#)
16. [图解 Linux 操作系统架构设计与实现原理（第二版）](#)
17. [如何在 Ubuntu 和 Linux Mint 下轻松升级 Linux 内核](#)
18. [Android 简单 mp3 播放器源码](#)
19. [嵌入式 Linux 系统实时性的研究](#)
20. [Android 嵌入式系统架构及内核浅析](#)
21. [基于嵌入式 Linux 操作系统内核实时性的改进方法研究](#)

## Windows CE:

1. [Windows CE.NET 下 YAFFS 文件系统 NAND Flash 驱动程序设计](#)
2. [Windows CE 的 CAN 总线驱动程序设计](#)
3. [基于 Windows CE.NET 的 ADC 驱动程序实现与应用的研究](#)
4. [基于 Windows CE.NET 平台的串行通信实现](#)
5. [基于 Windows CE.NET 下的 GPRS 模块的研究与开发](#)
6. [win2k 下 NTFS 分区用 ntldr 加载进 dos 源代码](#)
7. [Windows 下的 USB 设备驱动程序开发](#)
8. [WinCE 的大容量程控数据传输解决方案设计](#)
9. [WinCE6.0 安装开发详解](#)
10. [DOS 下仿 Windows 的自带计算器程序 C 源码](#)
11. [G726 局域网语音通话程序和源代码](#)
12. [WinCE 主板加载第三方驱动程序的方法](#)
13. [WinCE 下的注册表编辑程序和源代码](#)
14. [WinCE 串口通信源代码](#)
15. [WINCE 的 SD 卡程序\[可实现读写的源码\]](#)
16. [基于 WinCE 的 BootLoader 研究](#)

## PowerPC:

1. [Freescale MPC8536 开发板原理图](#)
2. [基于 MPC8548E 的固件设计](#)
3. [基于 MPC8548E 的嵌入式数据处理系统设计](#)
4. [基于 PowerPC 嵌入式网络通信平台的实现](#)
5. [PowerPC 在车辆显控系统中的应用](#)
6. [基于 PowerPC 的单板计算机的设计](#)
7. [用 PowerPC860 实现 FPGA 配置](#)

## ARM:

1. [基于 DiskOnChip 2000 的驱动程序设计及应用](#)
2. [基于 ARM 体系的 PC-104 总线设计](#)

RT Embedded <http://www.kontronn.com>

3. [基于 ARM 的嵌入式系统中断处理机制研究](#)
4. [设计 ARM 的中断处理](#)
5. [基于 ARM 的数据采集系统并行总线的驱动设计](#)
6. [S3C2410 下的 TFT LCD 驱动源码](#)
7. [STM32 SD 卡移植 FATFS 文件系统源码](#)
8. [STM32 ADC 多通道源码](#)
9. [ARM Linux 在 EP7312 上的移植](#)
10. [ARM 经典 300 问](#)
11. [基于 S5PV210 的频谱监测设备嵌入式系统设计与实现](#)
12. [Uboot 中 start.S 源码的指令级的详尽解析](#)
13. [基于 ARM9 的嵌入式 Zigbee 网关设计与实现](#)

## Hardware:

1. [DSP 电源的典型设计](#)
2. [高频脉冲电源设计](#)
3. [电源的综合保护设计](#)
4. [任意波形电源的设计](#)
5. [高速 PCB 信号完整性分析及应用](#)
6. [DM642 高速图像采集系统的电磁干扰设计](#)
7. [使用 COMExpress Nano 工控板实现 IP 调度设备](#)
8. [基于 COM Express 架构的数据记录仪的设计与实现](#)
9. [基于 COM Express 的信号系统逻辑运算单元设计](#)