

文章编号: 1671-251X(2010)03-0068-04

# 基于 μC/OS-II 的 Ethernet 到 CAN 总线透明网关的研究

耿蒲龙<sup>1</sup>, 徐杰<sup>2</sup>

(1. 太原理工大学电气与动力工程学院, 山西 太原 030024;

2. 煤炭科学研究院常州自动化研究院, 江苏 常州 213015)

**摘要:** 针对 CAN 总线和 Ethernet 之间的互联互通问题, 设计了一种以微控制器 AT91M40800 为核心的 Ethernet 到 CAN 总线透明网关。该网关采用网卡芯片 RTL8019AS 构成 Ethernet 接口电路; 采用微控制器+CAN 接口控制芯片 SJA1000+CAN/接口收发芯片 PCA82C250 结构模式构成 CAN 总线接口电路; 在 μC/OS-II 操作系统运行平台上嵌入了轻量级 TCP/IP 协议栈 LwIP, 在网关具备了 Ethernet 通信功能和标准 CAN 节点运行功能的基础上将协议转换设置成透明传输方式。测试结果表明, 该网关实现了 CAN 总线与 Ethernet 之间的互联互通。

**关键词:** 网关; CAN 总线; Ethernet; 透明传输; μC/OS-II; AT91M40800; LwIP

**中图分类号:** TN915.05      **文献标识码:** A

## 0 引言

现场总线和 Ethernet 技术在工矿企业的现场设备层和信息管理层得到了越来越广泛的应用。要实现从控制层到管理层、决策层的信息交换和集成, 达到 Ethernet 和现场总线 2 种异质网络之间的无缝链接, 就必须使用协议转换装置。本文针对现场总线中使用较广泛的 CAN 总线和 Ethernet 之间的互联互通问题, 设计了一种基于 μC/OS-II 的透明网关, 该网关具有成本低、实时性高等优点。

收稿日期: 2009-11-12

基金项目: “十一五”国家科技支撑计划重点项目  
(2007BAK29B05)

作者简介: 耿蒲龙(1982-), 男, 山西临汾人, 助教, 现从事矿用智能电器技术方面的研究及教学工作。E-mail: gpl\_1016@163.com

- [2] 王俊林, 张剑云. 基于统计证据的 Mass 函数和 DS 证据理论的多传感器目标识别[J]. 传感技术学报, 2006(3): 863-865.
- [3] 朱大奇, 于盛林. 故障诊断的信息融合算法比较分析[J]. 控制理论与应用, 2004(4): 662-663.
- [4] LIN J. Feature Extraction of Machine Sound Using Wavelet and Its Application in Fault Diagnosis [J]. NDT & E International, 2001(34): 25-30.
- [5] REN Zhen, HE Jianjun. Compression and Recons-

## 1 硬件设计

基于 μC/OS-II 的 Ethernet 到 CAN 总线透明网关的核心处理器选用 ATMEL 公司的 ARM7 微控制器 AT91M40800<sup>[1]</sup>, 因其内部只集成了 8 KB 的 SRAM, 为了给程序运行和数据存储提供空间, 扩展了存储器模块; 根据网关功能要求设计了 Ethernet 接口和 CAN 总线接口电路; 键盘和显示接口电路则构成了人机通道; 此外, 该网关在线调试使用了 JTAG 电路。其硬件结构如图 1 所示。

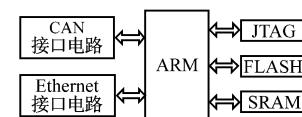


图 1 基于 μC/OS-II 的 Ethernet 到 CAN 总线透明网关硬件结构

truction Faults for Faults Signals of Electric Machines Based on Wavelet Packets [J]. Proceedings of the CSEE, 2001, 21(1): 25-29.

- [6] 黄建. 基于小波分析的变频调速系统故障诊断技术的研究[D]. 焦作: 河南理工大学, 2008.
- [7] GAO H Y, BRUCE A G. Wave Shrink with Firm Shrinkage [J]. Statist, 1997, 7(4): 855-874.
- [8] 刘振兴, 尹项根, 张哲. 基于瞬时功率信号频谱分析的鼠笼式异步电动机转子故障在线诊断方法[J]. 中国电机工程学报, 2003(10): 150-151.

## 1.1 Ethernet 接口电路

Ethernet 接口电路主要由 RTL8019AS 网卡芯片、20F001 网络变压器以及 RJ45 接头构成。该接口电路较为成熟,在此不做介绍。

## 1.2 CAN 总线接口电路

CAN 总线接口电路采用了较为常见的微控制器+ CAN 接口控制芯片 SJA1000+ CAN 接口收发芯片 PCA82C250 结构模式,如图 2 所示。为了提高系统的抗干扰能力,在 SJA1000 和 PCA82C250 之间采用了高速光耦 6N137。由于 SJA1000 为地址数据复用结构,而 AT91M40800 为地址、数据总线分开的结构,二者不能直接相连,笔者通过在二者之间加入可编程逻辑器件 GAL16V8D 来解决 AT91M40800 和 SJA1000 之间的时序匹配问题。

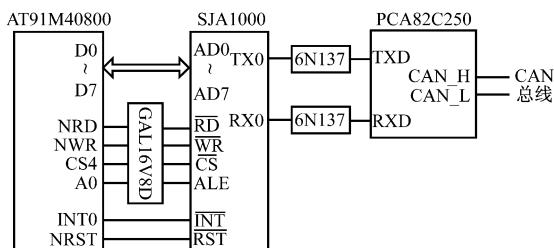


图 2 CAN 总线接口电路

GAL16V8D 的输入信号为 AT91M40800 的读(NRD)、写(NWR)、片选(CS4)和 A0,输出信号为 AT91M40800 访问 SJA1000 所需要的读、写、片选及地址锁存信号。输入输出信号的组合逻辑设计思想如下: AT91M40800 访问 SJA1000 所需要的 ALE 信号由 AT91M40800 的 NWR 信号取反后和 CS4、A0 相与得到; SJA1000 的写信号则由 AT91M40800 的 NWR 和 A0 相或产生; SJA1000 的读信号则由 AT91M40800 的 NRD 和 A0 进行或逻辑产生; SJA1000 的片选信号由 AT91M40800 的 CS4 取反后与 A0 相或得到。

使用 VHDL 语言在 ispLEVER 开发环境中实现上述组合逻辑,对该组合逻辑进行仿真得到的时序如图 3 所示。

根据该组合逻辑,当 AT91M40800 对 SJA1000 进行写操作时,应先向所定义的地址端口写 1 次地址值,再向所定义的数据端口写 1 次数据值,即 AT91M40800 要通过 2 次写操作才能完成真正意义上对 SJA1000 的 1 次写操作。而当 AT91M40800 对 SJA1000 进行读操作时则是先向

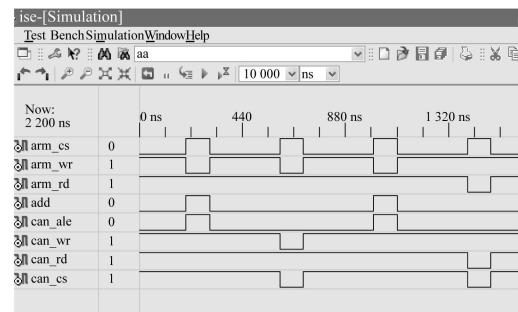


图 3 GAL16V8D 中的组合逻辑时序仿真

所定义的地址端口写 1 次地址值,再从所定义的数据端口读 1 次数据值,即 AT91M40800 需要通过 1 次写操作和 1 次读操作来完成真正意义上对 SJA1000 的 1 次读操作。

笔者将测试值先写入 SJA1000 的测试寄存器(CAN 总线地址为 09H),再读出该寄存器的值,通过判断读、写值的一致性来验证所设计的组合逻辑电路能否正确工作。实测结果表明,写入寄存器中的测试值和读出值相同,CAN 总线接口时序逻辑电路正确。

## 2 软件设计

基于 μC/OS-II 的 Ethernet 到 CAN 总线透明网关的软件设计包括 μC/OS-II 操作系统移植<sup>[2]</sup>、轻量级 TCP/ IP 协议栈 LwIP 的嵌入、CAN 通信程序设计以及协议转换程序的设计,本文主要介绍后 3 个部分内容。

### 2.1 LwIP 的嵌入

传统的 TCP/ IP 协议的实现过于复杂,需要占用大量系统资源,还不能满足嵌入式系统的实时性要求。本文选用了轻量级 TCP/ IP 协议栈 LwIP 作为协议转换器的 TCP/ IP 协议栈<sup>[3]</sup>。LwIP 在设计时就已考虑了将来的移植问题,将所有与硬件、OS 和编译器相关的部分独立出来,放在\src\arch 目录下。LwIP 的嵌入就是修改该目录下的文件,主要包括操作系统模拟层的修改和 LwIP 网络接口驱动函数编写 2 个部分内容,如图 4 所示。

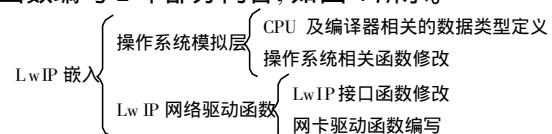


图 4 实现 LwIP 嵌入的主要内容

其中,操作系统相关函数修改主要包括信号量、邮箱、创建线程、超时机制等函数的实现; LwIP 接口函数修改则是指实现 LwIP 源代码 ethernetif.c

文件中的 `ethernetif_init()`、`ethernetif_output()`、`ethernetif_input()`、`low_level_init()`、`low_level_output()`、`low_level_input()` 六个函数；网卡驱动函数编写则是根据本文所使用的网卡芯片 RTL8019AS 编写了初始化及收、发包驱动函数。

进行 LwIP 嵌入测试时，将网关与台式机经网线相连，RTL8019AS 的 MAC 地址设置为 0x00、0x16、0x36、0x7D、0x94、0xC8；IP 地址设置为 192.168.100.175。台式机 IP 地址设置如图 5 所示。在命令行窗口运行 ping、arp 指令，结果如图 6 所示，这说明 LwIP 协议栈运行正常，协议栈嵌入成功。

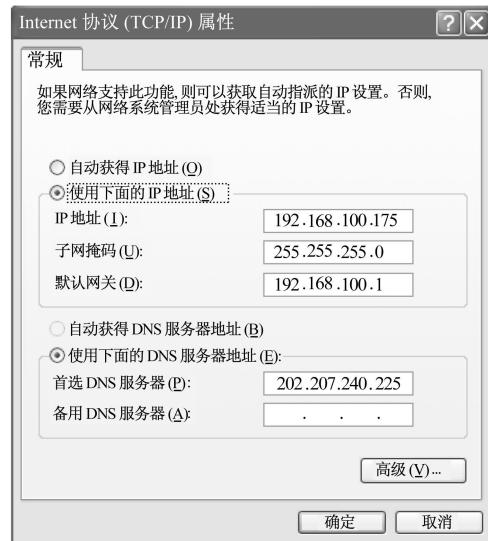


图 5 台式机 IP 地址设置界面

```
Microsoft Windows XP [版本 5.1.2600]
(c) 版权所有 1985-2001 Microsoft Corp.

c:\Documents and Settings\Administrator>ping 192.168.100.175
Pinging 192.168.100.175 with 32 bytes of data:
Reply from 192.168.100.175: bytes=32 time=21 ms TTL=128
Reply from 192.168.100.175: bytes=32 time=5 ms TTL=128
Reply from 192.168.100.175: bytes=32 time=5 ms TTL=128
Reply from 192.168.100.175: bytes=32 time=5 ms TTL=128

Ping statistics for 192.168.100.175:
    Packets: Sent = 4, Received = 4, Lost = 0% loss<,
    Approximate round trip times in milli-seconds:
    Minimum = 5 ms, Maximum = 21 ms, Average = 9 ms

c:\Documents and Settings\Administrator>arp -a
Interface: 192.168.100.170 --- 0x2
  Interface Address      Physical Address      Type
  192.168.100.175      00-16-36-7d-94-c8  dynamic

c:\Documents and Settings\Administrator>
```

图 6 ping、arp 指令运行结果界面

## 2.2 CAN 总线接口驱动程序

CAN 总线物理层和数据链路层功能由 SJA 1000 的驱动程序和硬件自身功能共同实现。CAN 总线接口驱动程序主要包括 CAN 控制器的初始化、发送和接收程序。

### (1) CAN 控制器的初始化程序

在 SJA 1000 正常工作之前，微控制器需要通过

配置 SJA 1000 中的一些寄存器以实现对 SJA 1000 的初始化，主要完成对时钟分频寄存器(CDR)、中断使能寄存器(IER)、总线定时器(BTR0/1)、输出控制寄存器(OCR)、验收代码寄存器(ACR)和验收屏蔽寄存器(AMR)等的设置。需要注意的是，设置这些寄存器时应先使 SJA 1000 处于复位模式，设置完成后使 SJA 1000 进入正常工作模式。

### (2) CAN 控制器的发送程序

CAN 控制器的发送程序由上层程序调用，其流程如图 7 所示。

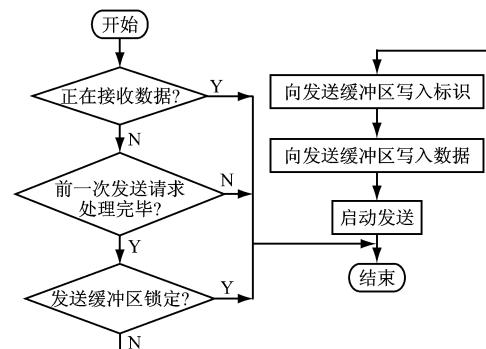


图 7 CAN 控制器的发送程序流程

### (3) CAN 控制器的接收程序

CAN 控制器的接收程序以中断方式调用，通过判断中断寄存器 (IR) 来决定是否进行数据接收操作，其流程如图 8 所示。

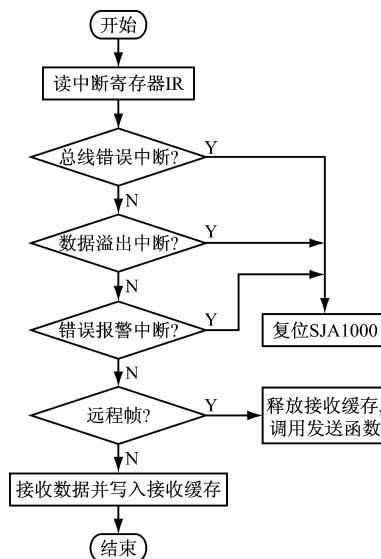


图 8 CAN 控制器的接收程序流程

通过将该网关与武汉吉阳光电生产的 GY8508 USB-CAN 适配器(标准 CAN 节点)通信来判断网关的 CAN 总线接口电路能否正常工作，测试程序流程如图 9 所示。

运行上述测试程序，实测到网关的 CAN 总线

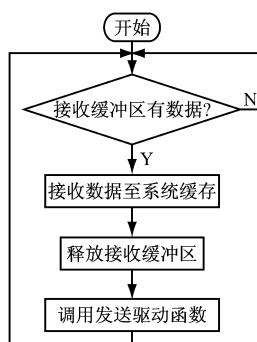


图9 网关的CAN总线通信测试程序流程

通信数据和GY8508 USB-CAN适配器的数据如图10所示,比较两者的收发数据可知收发数据相同,CAN总线接口电路工作正常,该网关具备了标准CAN节点运行能力。

该界面展示了CAN总线通信数据,包括帧类型、帧ID、CAN通道、数据长度、数据等信息。数据表中列出了多条收发报文,显示了CAN总线通信的详细数据。

	CAN通道	传输方向	ID号	帧类型	帧格式	数据长度	数据
0233:046	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0233:409	0	接收	00000000	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0233:750	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0233:843	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0234:546	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0234:640	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0235:187	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0235:231	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0235:475	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0235:598	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0236:687	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0236:781	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0237:687	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0237:781	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B
0238:718	0	发送	00000000	数据帧	扩展帧	8	11 22 33 44 55 66 77 88
0238:812	0	接收	00010200	数据帧	扩展帧	8	04 05 06 07 08 09 0A 0B

图10 网关的CAN总线通信数据和GY8508 USB-CAN适配器的数据界面

### 2.3 协议转换程序设计

在Ethernet和CAN总线之间传输数据时,笔者采用透明数据传输方式,也就是在Ethernet应用层构建和解析完整的CAN协议数据包,CAN协议数据包作为TCP/IP网络应用层的数据进行传输,它对通信数据的具体意义不做任何解释。采用透明数据传输方式,在硬件上无需扩展大容量的数据存储器用于缓存数据;软件上也不用考虑数据的聚集和分发处理,降低了程序设计的复杂程度、提高了通信的速度和实时性;透明数据传输方式可以应用于不同的项目要求;此外由于协议转换设计简单,没有较多的复杂处理,从而提高了通信的可靠性<sup>[4]</sup>。

该网关能将从Ethernet接收到的TCP/IP协议数据包转换为CAN协议形式发送至CAN网络中的指定节点,完成从Ethernet到CAN网络的数据传输。笔者创建了CAN\_TCP\_Task()任务完成上述Ethernet上的数据传输。反过来当CAN总线

上的数据要传输到Ethernet时,首先由CAN接收中断触发,中断函数给CAN\_FieldBus\_Task()任务发送信号量。该任务首先调用CAN控制器的接收程序,完成CAN数据的接收,再将CAN数据封装成TCP/IP协议的Ethernet帧,最后发送到Ethernet上。这2个任务的流程如图11所示。

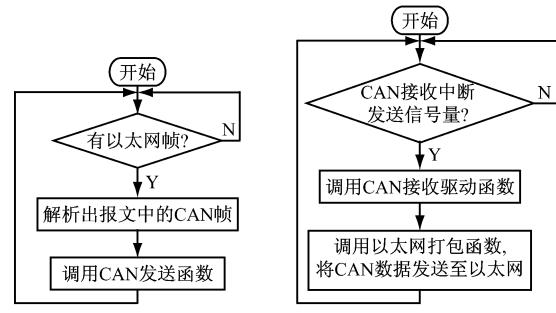


图11 CAN\_TCP\_Task()和CAN\_FieldBus\_Task()任务的流程

### 3 结语

本文针对现场总线中使用较广泛的CAN总线和Ethernet之间的通信互联问题,介绍了透明网关的设计,采用了软、硬件资源丰富的网卡芯片RTL8019AS构成网关的Ethernet接口电路,降低了设计难度;通过在可编程逻辑器件中进行组合逻辑电路设计,解决了AT91M40800和SJA1000之间的时序配合问题;在网关具备了Ethernet通信功能和标准CAN节点运行功能的基础上将协议转换设计成透明传输方式,简化了系统设计难度,提高了系统运行的实时性和可靠性。

### 参考文献:

- [1] 朱义军,杨育红,赵凯,等.AT91系列ARM微控制器体系结构与开发实例[M].北京:北京航空航天大学出版社,2005.
- [2] 吴鹏,傅成华,干树川.CAN总线接口电路设计[J].中国西部科技,2009,8(13):51-53.
- [3] LABROSSE J J.嵌入式实时操作系统μC/OS-II[M].2版.邵贝贝,译.北京:北京航空航天大学出版社,2003.
- [4] DUNKELS A. Design and Implement of the LwIP TCP/IP Stack [EB/OL]. (2001-02-20). <http://savannah.nongnu.org/projects/lwip>.
- [5] 袁学文.CAN总线与Ethernet互联系统设计[D].武汉:武汉大学,2004.