

# 基于 SJA1000 的 CAN 总线控制节点的设计与实现

李 云<sup>1</sup>, 王宝良<sup>2</sup>, 丁在田<sup>2</sup>, 王延伟<sup>2</sup>

(1. 海军工程大学, 湖北 武汉 430000; 2. 空军工程大学 电讯工程学院, 陕西 西安 710077)

**摘 要:**介绍了 CAN 总线的特点以及一种新型的独立 CAN 控制器 SJA1000 的主要功能, 给出了一种基于 SJA1000 的 CAN 控制节点的硬件电路及软件设计, 并对设计中应注意的问题进行了较详细的说明。

**关键词:** CAN 总线; 控制节点; SJA1000

**中图分类号:** TP393.1    **文献标识码:** A    **文章编号:** 1009-3516(2003)03-0079-04

典型的 CAN 总线控制节点是由微处理器、CAN 控制器、CAN 接口构成。多个控制节点通过 CAN 接口挂接到 CAN 总线上, 各控制节点都有权向其它控制节点发送信息, 形成多主机控制器局域网。由于 CAN 总线是基于发送报文的编码, 而不是对 CAN 控制节点进行编码, 所以增加或删除总线上的控制节点不会对系统造成太大的影响。且 CAN 节点在错误严重的情况下, 具有自动脱离总线的功能, 能切断自身与总线的联系, 而使总线上其它节点不受影响。CAN 总线的这种独特的联结方式, 使得它与传统的串行通信方式相比, 具有诸多优点。

## 1 基于 SJA1000 的控制节点设计

目前广泛流行的 CAN 总线器件分为 2 大类, 即独立 CAN 控制器和带有在片 CAN 的微控制器(见图 1)。SJA1000 是 PHILIP 公司生产的一种新型独立 CAN 控制器, 是 PCA82C200 的替代产品, 它在软件的引脚上都与 PCA82C200 兼容。SJA1000 具有 BasicCAN 和 PeliCAN 2 种工作方式, 这 2 种工作方式的区别是: 在 PeliCAN 方式下, SJA1000 有一个重新设计的含有很多新功能的寄存器组, 且包含了 PCA82C200 中的所有位, 同时增加了一些新的功能位。PeliCAN 支持具有很多新特性的 CAN2.0B 协议。在 PeliCAN 模式下, SJA1000 的主要新功能有: ①标准结构和扩展结构信息的接收和传递; ②64 Byt 的接收 FIFO; ③在标准和扩展式中都有单\双接收过滤器; ④读\写访问的错误计数器; ⑤可编程的错误限制报警; ⑥最近一次的误码寄存器; ⑦对每一个 CAN 总线错误的错误中断; ⑧由功能位定义的仲裁丢失中断; ⑨一次性发送(当错误或仲裁丢失时不重发); ⑩只听模式(CAN 总线监听, 无应答, 无错误标志)、支持热插(无干扰软件驱动位速检测)、硬件禁止 CLKOUT 输出等。

### 1.1 节点硬件电路设计

控制节点以 8 位单片机 AT89C51 为核心, 选用 SJA1000 作为 CAN 控制器, 并使用了 CAN 控制器接口芯片 82C250 和光耦 HCPL2630, 硬件电路如图 2 所示。图中 AT89C51 单片机的 ALE、WR、RD 端分别控制 SJA1000 的 ALE/AS、WR、RD 端, 地址和数据线 AD<sub>0</sub>~AD<sub>7</sub> 由 PO 口分时复用实现。SJA1000 的中断请求信号 INT, 在中断允许且有中断发生时, INT 由高电平跳变到低电平, 所以 INT 和 AT89C51 的 INT0 直接相连, 片选信号 CS 由 P2.0 控制, 当 CS 接到低电平时, SJA1000 被选中, CPU 可对 SJA1000 进行读/写操作。

收稿日期: 2002-11-08

基金项目: 军队科研基金资助项目(KJ01205)

作者简介: 李 云(1962-), 女, 江苏苏州人, 副教授, 主要从事电路与系统研究。

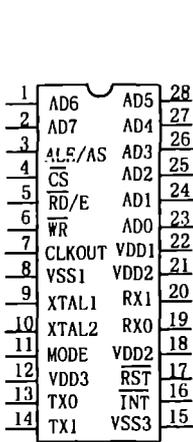


图1 SJA1000 管脚图

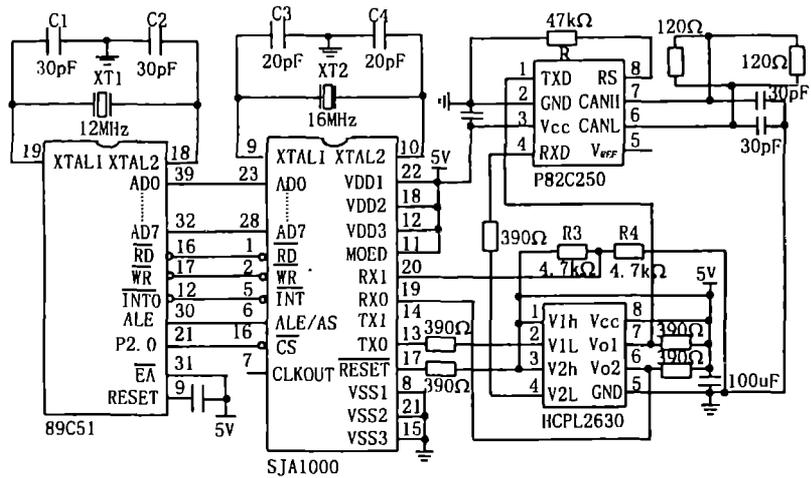


图2 节点硬件电路图

为了增强控制节点的抗干扰能力,防止线路间串扰,SJA1000 通过光耦 HCPL2630 与 P82C250 相连,从而使总线上各个 CAN 节点之间实现隔离,以保护 CAN 控制器。

82C250 是 CAN 总线收发器,是 CAN 控制器 SJA1000 与 CAN 总线的接口器件,对 CAN 总线以差分式发送。其引脚 RS 用于选择 82C250 的工作模式为高速、斜率控制或等待。RS 脚接地,82C250 工作于高速方式;RS 脚串接一个电阻 R 后再接地,用于控制上升和下降斜率,以减少射频干扰,实验数据表明 R 在 (15 ~ 200)kn 范围内取值较为理想。RS 脚接高电平,82C250 处于等待工作方式,此时发送器关闭,接收器处于低电流工作状态,可以对总线上的显性位作出反应。若 82C250 处于 CAN 总线的网络终端,总线接口部分必须加一个 120 Ω 的匹配电阻,以保护 82C250 免受过流的冲击。

1.2 节点软件设计

CAN 节点的软件设计主要包括 3 部分:SJA1000 初始化,报文的发送和接收。对 SJA1000 的初始化是对工作方式、接收滤波寄存器、接收屏蔽寄存器、接收代码寄存器、波特率参数等的设置,要求对写入每个寄存器的内容进行仔细和全面的考虑。对 SJA1000 初始化成功后,就可以用它来传输报文。节点向总线发送报文的过是:单片机 AT89C51 将待发送的数据按 CAN 格式组成一帧报文,写入 SJA1000 的发送缓冲区,然后启动 SJA1000,把报文发送到总线上。接收报文的过是:SJA1000 从总线上自动接收报文,并经过过滤后存入接收缓冲区,并向单片机发出中断请求,单片机从缓冲区读取报文。

根据硬件电路,用 MCS-51 汇编语言编写的初始化程序和发送报文字程序如下:(首地址为 0E00H):

1) 初始化子程序

```

CR EQU 0E00H ; 控制寄存器 CR
CMR EQU 0E01H ; 命令寄存器 CMR
SR EQU 0E02H ; 状态寄存器 SR
IR EQU 0E03H ; 中断寄存器 IR
ACR EQU 0E04H ; 接收码寄存器 ACR
AMR EQU 0E05H ; 接收屏蔽寄存器 AMR
BTR0 EQU 0E06H ; 总线定时器寄存器 0 BTR0
BTR1 EQU 0E07H ; 总线定时器寄存器 1 BTR1
OCT EQU 0E08H ; 输出控制寄存器 OCR
CDR EQU 0E1FH ; 时钟分频寄存的 CDR
    
```

```

MAIN: MOV DPTR, #CR ; 开始初始化
      MOV A, #01H
      MOVX @DPTR, A ;关所有 CAN 中断,置位复位请求
      MOV DPTR, #ACR ;
      MOV A, #80H
    
```

```

MOVX @ DPTR ,A ;置接收代码寄存器初值为 80H
INC DPTR
MOV A ,#7FH ;
MOVX @ DPTR ,A ;置接收屏蔽寄存器初值为 7FH
INC DPTR
MOV A ,#7FH
MOVX @ DPTR ,A ;BTR0 位速率为 5Kbps
INC DPTR
MOV A ,#7FH
MOVX @ DPTR ,A ;BTR1 位速率为 5Kbps
INC DPTR
MOV A ,#0AAH ;
MOVX @ DPTR ,A ;输出控制寄存器处于正常输出模式
MOV DPTR ,#CDR
MOV A ,#48H
MOVX @ DPTR ,A ;CLOCK OUT 频率为 8MHZ
MOV DPTR ,#CR
MOV A ,#1EH
MOVX @ DPTR ,A ;初始化结束,清除复位请求,SJA1000 进入正常工作状态。

```

## 2) 发送子程序

```

Loop1 : MOV DPTR ,#SR ;
        MOVX A ,@ DPTR ;从 SJA1000 读入 SR 的值
        JB ACC.4 ,WAIT1 ;判断是否正在接收,正在接收则等待
Loop2 : MOVX A ,@ DPTR
        JNB ACC.2 ,Loop2 ;判断发送缓冲区是否锁定,锁定则等待
        MOV DPTR ,TXB ;送发送缓冲区首地址
        MOV A ,#88H ;发送数据长度为 8 个
        MOVX @ DPTR ,A
        INC DPTR
        MOV A ,#ID0 ;送 4 个字节标识符(ID0 ~ ID3)
        MOVX @ DPTR ,A ;送 4 个字节标识符(ID0 ~ ID3)
        |
        | 重复以上三步直到送完 ID3
        MOV R0 ,#TRDATA ;送 CPU 发送数据区首地址
Loop3 : MOV A ,@ R0
        INC DPTR
        MOVX @ DPTR ,A
        INC R0
        CJNE R0 ,#TRDATA +8 ,Loop3 ;向发送缓冲区写 8 个字节
Loop4 : MOV DPTR ,#SR
        MOV A ,@ DPTR
        JNB ACC.3 ,Loop4 ;判断上次发送是否完成,未完成则等待
        MOV DPTR ,#CMR
        MOV A ,#05H
        MOVX @ DPTR ,A ;启动 SJA1000 发送
        RET

```

### 1.3 软件设计中应注意的几个问题

1) SJA1000 有 BasicCAN 和 PeliCAN 2 种工作方式,这 2 种工作方式可通过时钟分频寄存器中的 CAN 方式位来选择。上电复位默认工作方式为 BasicCAN 方式。在复位状态下,即只有当控制寄存器 CR 中的复位请求位 CR.0 被置为“1”时,才可以访问 ACR,AMR,BTRO,OCR,BTR1,OCR,CDR 这些寄存器。

#### 2) 接收码寄存器 ACR 和接收屏蔽寄存器 AMR 的初始化

由于每个节点都可以发送报文,而每个发送到总线上的报文都供所有节点同时接收。但是接收并不等于接受,每个节点是否接受一个报文取决于该报文是否与自己相关,相关则接受,无关则不接受。这里 ACR 和 AMR 组成了验收过滤,控制节点通过验收过滤来决定是否接收总线上的报文。接收报文的条件:接收代码位(AC.7 ~ AC.0)和信息识别的高 8 位(ID.10 ~ ID.3)相等,且与接受屏蔽位(AM.7 ~ AM.0)的相应位相或为 1,则报文被接收。如在初始化过程中 ACR = 80H,AMR = 7FH,因此只有信息识别码的高 8 位为 1xxxxxxx 的数据帧被接收。

#### 3) 总线定时器 BTRO 和 BTR1 的初始化

总线定时器 BTRO 的值可决定波特率预分频器(BRP)和同步跳转宽度(SJW)的数值;BTR1 可决定位周期的宽度、采样点的位置以及在每个采样点进行采样的次数。所以总线定时器寄存器确定了时钟周期、系统周期、位周期三者之间的关系,它的设置会直接影响 CAN 性能的发挥。在程序中 SJA1000 晶振频率为 16 MHz,BTRO = 7FH,BTR1 = 7FH,则波特率为 5 kbps。必须说明的是,系统中所有节点的 BTRO 和 BTR1 的内容都必须应用相同,否则将无法进行通信。

## 2 结束语

我们已将所设计的 CAN 控制节点应用于短波电台集中遥控系统中,实现了 16 个遥控终端对 16 部短波电台的实施遥控通信。通过对 CAN 总线技术的应用,使多个控制模块通过 CAN 控制器挂接到总线上,形成多主机局部网,使遥控终端任选一部电台,一部遥控终端选多部电台,多部遥控终端同选一部电台进行通信,遥控距离可达 10 km,获得了满意的效果。

### 参考文献:

- [1] 邬宽明. CAN 总线原理和应用系统设计[M]. 北京:北京航空航天大学出版社,1996.
- [2] 王毅峰,李令奇. SJA1000 在数据采集与控制系统中的应用[J]. 微计算机信息,2001,17(4):16-18.
- [3] 陈杨,刘曙光,龙志强. 基于 CAN 总线的数据通信系统研究[J]. 测控技术,2000,19(10):53-55.
- [4] 吕永健,谢文俊,王瑾. 基于 PC/104 总线的飞机综合告警系统自动测试设备[J]. 空军工程大学学报(自然科学版),2002,3(2):24-26.

(编辑:门向生)

## Design and Realization of CAN Bus Control Node Based on SJA1000

LI Yun<sup>1</sup>, WANG Bao-liang<sup>2</sup>, DING Zai-tian<sup>2</sup>, WANG Yan-wei<sup>2</sup>

(1. Navy Force Engineering University, Wu'han, Hubei 430000, China 2. The Telecommunication Engineering Institute, Air Force Engineering University, Xi'an, Shaanxi 710077, China)

**Abstract:** This paper introduces characteristics of CAN Bus and the main functions of SJA1000, a new type of alone CAN controller. A kind of hard circuit and soft design of the CAN control node based on SJA1000 is presented. The problems that should be payed attention to in designing are discussed in details.

**Key words:** CAN bus; control node; SJA1000