

基于 CAN 总线的中控主机的系统设计

张 梁, 王景存, 梅 鏢

(武汉科技大学 信息科学与工程学院, 武汉 430081)

摘要: 智能中控主机作为多媒体会议室和多媒体展厅等现代化办公场所运转的控制核心, 其重要性不言而喻, 目前市面上主要采用的是基于 RS485 总线的传统组网模式, 导致系统的实时性较差、容错率较低、功能欠完善, 同时成本高昂; 针对这种情况, 在不允许改动被控设备的工作方式的条件下, 提出一种基于 CAN 总线网络, 以嵌入式系统作为主控设备, 以各功能驱动模块作为子设备的中控系统; 通过 CAN 总线控制器的优先级仲裁机制, 以及制定的带错误反馈的数据包通信协议, 实现了对各种被控设备的实时精准控制; 该中控主机系统硬件电路简单、功能模块方便扩展、成本低廉并且高效稳定; 通过实验测试, 所设计的智能中控主机性能良好, 达到了规定的行业标准, 具有广泛的应用前景。

关键词: 智能中控; 通信; 数据结构; CAN 总线

A Center Control Computer System Based on CAN Bus

Zhang Liang, Wang Jingcun, Mei Biao

(College of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China)

Abstract: As the control core of the modern office space, such as multimedia conference room and multimedia exhibition hall, the smart center control computer is very important, it is almost based on RS485 bus in current market. This will result in the worse real-time performance, the lower error-tolerant rate, the more deficient function and the higher price. In according with this situation, a smart center control computer system is introduced, which is based on CAN Bus network, with the embedded system as the master device, and the variant function modules as slave devices. Through the priority arbitration mechanism of CAN Bus controller, and the data packet communication protocol which contains error feedback, the real-time and accurate control to various controlled equipments is realized. So, the hardware circuit of the control system is simple, the function module is easy to expand and the cost is lower. According to the test, the design of the smart center control computer is good enough to meet the requirements of industry standards, and has a broad applications.

Keywords: smart center control computer; communication; data structure; CAN bus network

0 引言

随着互联网技术和信息通信技术的迅猛发展, 智能化、信息化渗透于人们生活的各个领域, 极大地改善了人们的生活环境和生活质量, 特别是在一些会议控制、演播和媒体演示、视频会议系统等领域, 高效率、低成本的中控系统显得尤为重要。中控主机是一套具备可再编程的嵌入式主机, 它提供各种常用的通信接口, 每个接口的作用都可以由用户定义, 接口之间的运作方式也可以通过再编程实现。接口的主要类型有串口、红外发射端口、数字 IO、USB 口、继电器、以太网、WiFi 等^[1]。

当前广泛使用的中控系统都使用 RS485 总线, 这使得各模块在通信时数据传输率较低, 容错率不高, 一个节点的通信错误可能会导致整个系统的瘫痪。鉴于此情况, 设计了一套基于 CAN 总线的中控主机系统。CAN 总线具有多主发送、采用确定性的优先级仲裁机制等特点, 保证了 CAN 总线数据通信的可靠性、实时性和灵活性^[2]。使得该系统有效地解决了上述问题, 极大地提升了系统的实时性和容错性。

1 系统组成

电气系统组成如图 1 所示。系统由 3 个模块组成: 外界交互模块、内部通信模块和功能驱动模块。外界交互模块用于接收用户 PC 或 IPAD 的命令请求并将命令处理后通过 CAN 总线传输给功能驱动模块响应。内部通信模块采用 CAN 总线作为连接各个功能驱动模块和核心板的纽带。功能驱动模块用于驱动各种电器设备。系统将各功能模块严格区分, 分工明确, 避免了单个 CPU 负载量过大的情况。

核心板由主频高达 806 MHz 的 Colibri XScale PXA320 组成, 运行在 WinCE 环境下, 具备多种通信方式的通信接口, 向用户提供人机友好的交互界面。红外子模块和继电器 IO 子模块选用飞思卡尔公司的 DZ32 单片机, 带有 CAN 模块接口。串口子模块需要 4 路独立串口, 选用飞思卡尔公司的 K10, 带有 6 个独立的 UART 模块。

中控主机工作在两种模式下: 初始化模式和正常工作模式。在初始化模式下, 核心板通过约定的数据帧结构对各功能模块进行初始化配置, 包括红外数据的烧写和串口波特率的配置等等。在正常工作模式下, 核心板通过约定的命令帧结构控制各功能模块。

2 硬件电路设计

2.1 CAN 总线接口电路

CAN 总线接口电路由 CAN 总线控制电路和 CAN 总线收发

收稿日期:2015-08-31; 修回日期:2015-09-25。

作者简介: 张 梁(1991-), 男, 湖北仙桃人, 硕士研究生, 主要从事数字信号处理及系统设计的研

王景存(1963-), 男, 山东菏泽人, 教授, 主要从事数字信号处理和计算机控制等方向的研究。

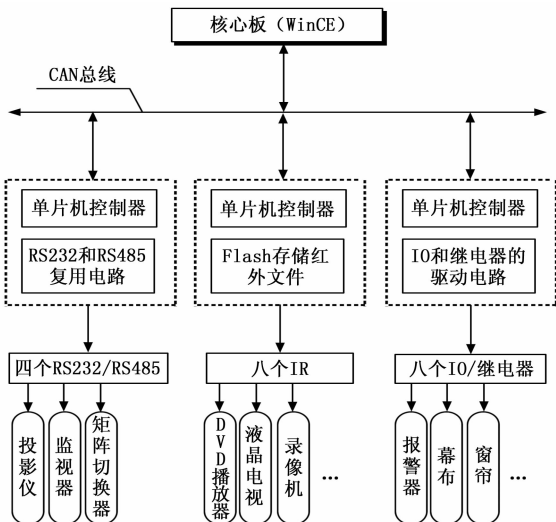


图 1 系统总体框图

电路两部分组成, 如图 2 所示。飞思卡尔的 K10 和 DZ32 都自带 CAN 总线控制电路, 只需连接 CAN 总线收发电路即可接入总线。为了避免芯片内的 CAN 控制模块和 CAN 收发电路的电气不一致造成电气隔离给通信带来麻烦, 模块在芯片和 CAN 收发电路之间添加 ADuM1301 隔离芯片。

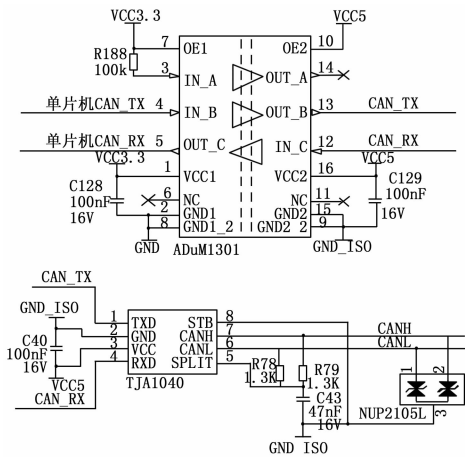


图 2 CAN 总线接口电路

ADuM1301 隔离芯片采用了 iCoupler 技术取消了光电耦合器中的光电转换, 该隔离芯片所隔离的两端有各自的电源和参考地, 使用了该芯片中 3 个通道的一对输入/输出, 适合 CAN 总线双向收发的特性。

CAN 总线收发器采用 TJA1040 收发器, 该收发器将用于传输的差分电压转换为显性/隐性状态提供给单片机。引脚 STB 选择工作模式, 本设计将其接地即为高速模式。在 CAN 总线接入端 CANH 和 CANL 与地之间分别接一个双向二极管, 能有效防止 ESD 和浪涌电压, 起到一定的过压保护作用。

2.2 RS232 和 RS485 复用电路

为了与具有串口的电气设备通信, 中控主机外置 4 个 RS232 串口, 但考虑到部分受控设备分布距离较远, 故需要采用 RS485 来解决长距离通信质量。本系统中采用单片机串口复用的方式, 根据需要实现单片机串口 TTL 电平与 RS232 电平的转换。当单片机串口 TXD 信号作为输出时, 可以直接接在

SP3232 和 SP3485 的 TTL 输入端, 因为两个收发器芯片的对应输入脚的输入电阻都很大, 实际相当于两个并联的大电阻, 完全可以接收 TXD 的信号, 转换后的电平输出也是正常的。但是当单片机串口 RXD 信号作为输入时, 不能直接接在 SP3232 和 SP3485 的 TTL 输出端, 因为两个收发器的 TTL 输出端是非开漏结构, 不能实现线与功能, 理论上, 单片机的 RXD 接收到的只能是驱动能力强的那个收发器信号, 如果两个收发器驱动能力差不多, 则识别的电平时有时无。为了实现线与功能, 将 SP3232 和 SP3485 的 TTL 输出端连接到两输入与门 NC7SZ08 上, 并且均上拉 4.7 K 电阻。

如图 3 所示, 单片机串口 UART1_TX 直接接在 SP3232 和 SP3485 的 TTL 输入端, 而 SP3232 的 TTL 输出端 232_1_RX 和 SP3485 的 TTL 输出端 485_1_RX 通过 NC7SZ08 线与后输出给单片机的 UART1_RX。UART1_RTS 和 UART1_CTS 是请求发送和清除发送信号, 接到单片机上。TX1_OUT、RX1_IN、RTS1_OUT 和 CTS1_IN 通过 DB9 连接外部电气设备。485_1_R/T 控制 SP3485 的数据接收/发状态, 接到单片机上。

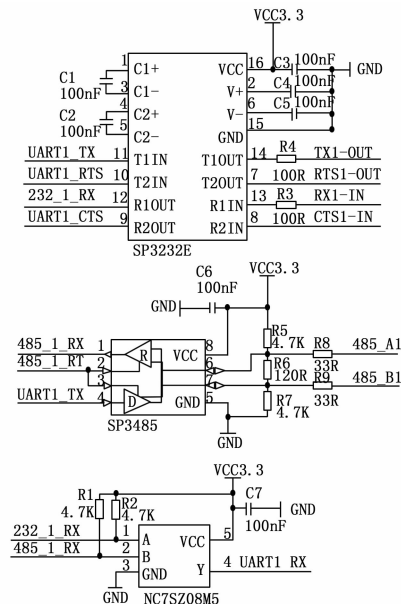


图 3 RS232/RS485 复用电路

2.3 红外发射电路

DZ32 单片机根据用户的命令查找到对应的红外键值码后, 由单片机 PWM 模块解码为 38K 的红外载波信号经 IO 口输出。IO 口输出的红外载波信号经 74HC138 译码器扩展为 8 路反向波形, 再通过 74HC240 反向驱动后即可使用。用户将该 8 路红外通过红外发射棒引出, 用于遥控各个红外接受设备。

由于红外文件是通过美国 SMARTISYS 公司的红外学习器按一定的协议学习后生成的, 生成的 sir 红外文件大小从几百 Byte 到十几 Kbyte 不等, 这是由遥控器的按键数量和一键多值共同决定的, 单片机的内部 Flash 容量显然不足。为了满足不同用户的实际需求, 选用 AT45DB041D 串行 Flash 存储芯片, 高达 4 Mbit (512 Kbyte) 存储空间, 共 2 048 页, 可以为每个红外文件分配 256 页 (约 67 Kbyte) 的存储空间。AT45DB041D 与单片机采用 SPI 串行通信, 最大支持 66 MHz, 满足实时性要

求，与单片机的连线非常简单，只需将主进从出引脚 SO，主出从进引脚 SI，时钟引脚 CLK，片选引脚 CS 连接到单片机的 4 个 IO 管脚上。该 Flash 采用 3.3 V 供电，复位管脚 $\overline{\text{RESET}}$ 和保护管脚 $\overline{\text{WP}}$ 接到 3.3 V 上，即可通信。

2.4 双向 IO 和继电器驱动电路

双向 IO 电路非常简单，从单片机引出的 8 个 IO 经 74ABT573 八进制 D 型锁存器后三态输出，外部输入的 8 路 IO 经 74HC240 反向输入至单片机的 8 个 IO 上。

继电器驱动电路是将单片机的 8 路 IO 经 SN74LS373 锁存器后，再通过 ULN2801A 八个达林顿阵列获取更高的电流驱动能力驱动八个 DS2Y-S-DC5V 双刀双掷继电器。

3 软件设计

3.1 数据结构

本设计在 CAN 总线上以组合包（一个包头+一个附加数据段）的格式通信，当没有附加数据段时，组合包就是包头。组合包分为上行组合包（由于模块反馈给核心板）和下行组合包（核心板发送给子模块的命令帧或扩展帧）。为了更好的代码维护，上行组合包和下行组合包采用相同的格式^[3]。其中包头格式如下所示：

```
typedef struct uart_can_msg_head
{
    unsigned char FrameHead; // 帧头, 设定为 0xAA
    unsigned char bFrameType; // 0 表示命令帧; 1 表示扩展帧
    unsigned char DeviceId[2]; // 发送 ID
    unsigned char DevicePort; // 模块端口号
    unsigned char PortIndex; // 对应模块的索引号
    unsigned char DataType; // 数据类型, 红外(0x01)、串口(0x02)、继电器(0x03)
    unsigned char DataLength[2]; // 附加数据长度
    unsigned char Reserve[5]; // 保留字段
    unsigned char CheckSum; // 校验和
    unsigned char FrameTail; // 帧尾, 设定为 0xAB
};
```

帧头和帧尾规定了一个包头的开始和结束。模式用来区分正常工作模式和初始化配置模式。发送 ID 用来区分各个功能模块。端口号和索引号是指向确定模块的具体端口。数据类型用于与前面指定的发送 ID 做双重判断。附加数据段长度指明附加段的字节个数，本设计将附加数据段放在 16 字节包头的后面，以组合包格式传输。保留字段表示组合包的个数。校验和由所有数据求和取反后加 1 获得，接收方只需将接受的数据（包括校验和）求和，结果为 0 则正确，如果有附加数据段，则包头和附加数据段一起校验，否则只校验包头。

3.2 CAN 通信模块的软件设计

CAN 通信模块的软件设计主要包括 CAN 的初始化程序和收发程序。CAN 的初始化包括波特率和滤波器的设置，其中波特率的设置公式如下：

$$CANbps = \frac{f_{CANCLK}}{(Pre_val) \cdot (1 + TimeSeg1 + TimeSeg2)}$$

其中：CANbps 是波特率， f_{CANCLK} 是振荡器频率，Pre_val 是预分频系数。TimeSeg1 和 TimeSeg2 分别是时间段 1 和时间段 2，1 代表位同步时间，3 个参数共同组成一个位时间，CAN

控制器在时间段 1 和时间段 2 的交界处点进行采样^[4]。

由于 CAN 协议本身只定义了物理层和数据链路层的规范，因此用户在设计 CAN 协议时必须根据实际需求自行定义 CAN 高层协议——CAN 通信应用层协议^[5]。在中控系统中，CAN 总线通信采用带 11 位报文标识符（ID 标识符）的标准帧和 16 位滤波器模式。为了在 16 位滤波器模式中接收标准标识符，需要把掩码寄存器 CANIDMR0/1 的低 11 位设置为 1（忽略匹配）或 0（严格匹配），考虑到中控主机功能模块的可扩展性，本设计匹配 3 位 ID，即可以挂接 8 台子模块。

当 CAN 的初始化完成之后，系统进入数据收发程序。按照 CAN 协议制定的优先级，首先安排接收或发送优先级高的数据。在 CAN 总线繁忙时，优先级低的接收或发送数据先存储于接收或发送缓存器中，等待 CAN 总线空闲的时候再自动重发。在设计中，采用最简单的基于标识符的优先级，即发送 ID 数值最小的获得最高的优先级。由于中控系统需要实时采集用户指令和处理数据，因此要求 CAN 网络通信具有实时性，这样 CAN 总线通信必然采用中断方式，在接受中断函数中提取出数据帧中的 8 字节数据。各功能模块采用相同的 CAN 接收中断处理模块，CAN 接收中断处理数据帧的细化流程如图 4 所示，下面做详细说明：进中断后手动匹配 ID，然后判断是否可以接受包头，如果可以，则对接受的包头进行帧标志符检查，然后判断其帧类型：扩展帧和命令帧。如果是命令帧，验证校验和接收即可；如果是扩展帧，立即禁止接受包头，防止下次进入中断时把附加段数据当做包头处理并在下次进入中断后接收附加段数据。

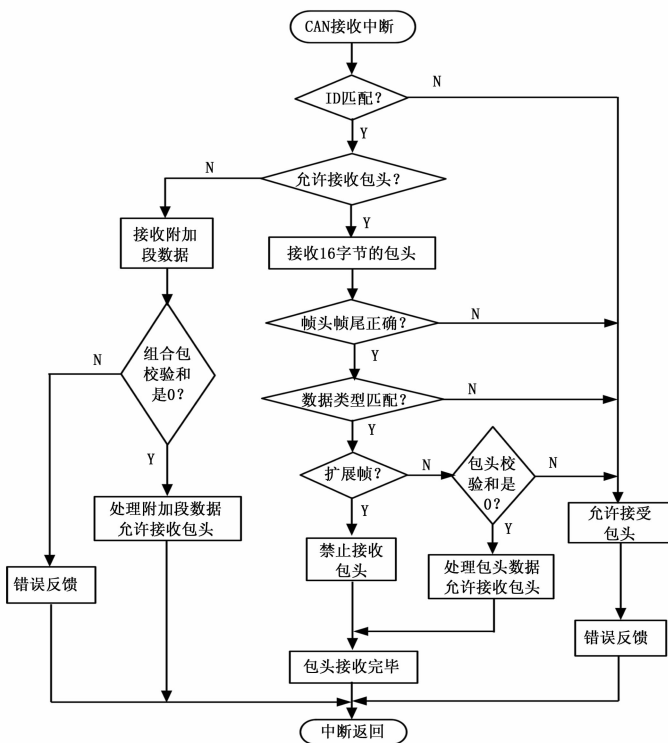


图 4 CAN 接收中断细化框图