

# 基于 ARM7 的 CAN 总线中继器的设计

余 涛,刘晓明,仲元宏

(重庆大学 通信工程学院,重庆 400044)

**摘 要:** CAN 总线是一种多主方式的串行通信总线。它具有抗干扰能力强、实时性高等特点,在工业上得到广泛应用。然而它的数据传输距离受传输速度的制约,并且同网络内节点数最大为 110。文中提出一种以 ARM7(LPC2119)为控制器的 CAN 总线中继的设计方法,以实现在不降低传输速度的情况下延长传输距离,同时进行两个网络间的数据转发并扩展网络中的节点数。该设计被应用在基于 CAN 总线的实时语音通信系统的运行过程中,运行稳定,无丢包现象,保证了系统的实时性。

**关键词:** CAN 总线;语音通信;CAN 控制器

**中图分类号:** TP336

**文献标识码:** A

**文章编号:** 1673-629X(2009)11-0198-04

## Design of CAN Repeater Based on ARM7

YU Tao, LIU Xiao-ming, ZHONG Yuan-hong

(College of Communication Engineering, Chongqing University, Chongqing 400044, China)

**Abstract:** CAN bus is multi-master mode serial communication bus. It has strong anti-interference ability, high real-time characteristic, and is widely used in the industry. However, CAN bus data transfer distance subject to constraints of transfer speed, and network with the largest number of nodes to 110. Present a design of CAN repeater based on ARM7(LPC2119), in order to achieve the extension of transmission distance, without lowering the speed of transmission at the same time to retransmit data between two networks and expand the nodes of network. This design is applied in a real-time voice communication system based on the CAN bus, it works stably, no data loss, and ensure the system's real-time performance.

**Key words:** CAN bus; voice communication; CAN bus controller

## 0 引 言

CAN 总线是一种多主方式的串行通讯总线并被公认为最有前途的现场总线之一<sup>[1]</sup>。然而由于受 CAN 收发器的限制, CAN 总线的最大的直接通信距离为 10km, 同网络中的节点数最多为 110 个<sup>[2]</sup>, 在某些场合, 这是不够的, 这时就需要用 CAN 总线中继器对 CAN 总线网络进行扩展。并且通过中继器还可以连接两个不同波特率的 CAN 总线网络, 在两个网络间进行数据转发<sup>[3]</sup>。

在基于 CAN 总线的实时数字语音通信系统中, 要求在保证通信距离的条件下总线上能同时通话的用户数越多越好。将 CAN 总线按照用户数分为多个时隙, 每个用户的数据占用一个时隙, 那么用户越多, 每个用户所占有的时隙越窄, 也就要求通信波特率越高, 然而

CAN 总线的波特率和通信距离是矛盾的, 传输距离越长, 允许的波特率越低, 因此, 为了在更高的波特率传输更远的距离, 在网络中接入 CAN 中继器。这样相当于在通信距离不变的条件下, 可以将波特率提高为原来的两倍, 即可以将用户数增加一倍。另外, 因为 CAN 中继器所连接的两个网络的节点在某时刻并不是每个节点都需要经过中继器与另一个网络中的节点进行数据交换, 因此, 在中继器中进行帧过滤, 需要转发的数据才转发到另一个网络中, 这同时也减轻了中继器的负担。

## 1 CAN 中继器硬件设计

系统总体硬件结构图如图 1 所示。微控制器选择 NXP 公司的 ARM7 芯片 LPC2119, 它内部集成了 2 个 CAN 控制器, 这样就不需要外接控制器, 从而使系统更简洁, 而且可靠。通过设置它的每个总线的数据波特率可达 1Mbps, 验收过滤器还为选择的标准标识符提供了 FullCAN 自动接收模式<sup>[4]</sup>。

收稿日期: 2009-02-26; 修回日期: 2009-05-04

**作者简介:** 余 涛(1983-), 男, 硕士研究生, 研究方向为嵌入式系统、汽车电子; 刘晓明, 教授, 博士后, 硕士研究生导师, 研究方向为汽车电子、测控技术、通信、图像处理。

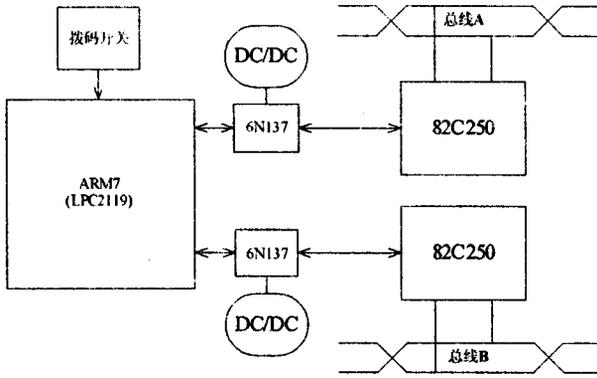


图 1 总体硬件结构图

其中一路 CAN 总线接口部分如图 2 所示,另一路与此类似。用高速光耦合器 6N137 将 CAN 控制器与收发器之间隔离开来,并且两路收发器都用独立的 DC/DC 供电,这样不仅实现了两路 CAN 接口之间的电器隔离,也实现了 CAN 中继器与总线之间的隔离。采取隔离措施,可以使故障局限在某一网段内,而不至于影响其他网段,既便于维护又保证了系统的安全。在 CAN 总线与 CAN 总线收发器 PCA82C250 之间加上了瞬态抑制二极管和自恢复保险丝,可有效地抑制总线上的尖峰脉冲,使系统更安全可靠<sup>[5]</sup>。120 欧终端匹配电阻为可选,如果网络中已经有一对匹配电阻,则不使用该电阻。

另外,为了应用于两个不同的波特率网络间的互连,外接拨码开关,可根据所要接入的网络的实际波特率来设置 CAN 中继器的相应端口波特率与之相适应。

## 2 CAN 总线中继器软件设计

CAN 中继器的主要任务是在两个网络中进行数据的过滤和转发。要系统能正常工作,LPC2119 的初始化和全局滤波设置是至关重要的。

全局接收过滤的工作流程:当 CAN 控制器的接收端已接收到一个完整的标识符,它将通知接收过滤器。接收过滤器响应这个信号,并读出控制器编号、标识符尺寸(11bit 或 29bit),然后,接收过滤器搜索 AF RAM (接受滤波 RAM)中的表格进行匹配,以决定接收或放弃这一帧信息<sup>[6]</sup>。

如果 FullCAN 模式使能,且 CAN 控制器告知当前信息包含一个标准标识符,则接收过滤器首先查询 FullCAN 标准标识符表格,以便接收可在 FullCAN 模式下处理。另外,如果 AF 未在 FullCAN 标准标识符表格中发现合适的匹配,它将接着依次查询下一个存在的标识符表格,以获取 CAN 控制器给出的标识符长度。只要发现匹配,接收过滤器就通知 CAN 控制器保存信息,并提供一个 ID 索引值使之保存到接收帧状态寄存器(CANRFS)中;若在所有存在的标识符表格中都未能得到匹配,AF 便通知 CAN 控制器丢弃/忽略接收到的信息<sup>[7]</sup>。从地址 0xE003 800087FF 为接收滤波 RAM,用于存放查找标识符表格 LUT 和 FullCAN 模式下自动接收 CAN 报文存储。0xE003 C000C01C 为接收过滤寄存器,用于控制过滤器工作模式及 LUT 格式。一个完整的接收过滤 RAM 内部表格分布如表 1 所示。表格的大小和个数可以根据实际过滤需要进行裁减<sup>[8]</sup>。

表 1 完整的接收过滤 RAM 表格

FullCAN 标识符表格
独立标准标识符表格
标准标识符范围表格
独立扩展标识符表格
扩展标识符范围表格
FullCAN 自动接收存储区

CAN 模块的初始化流程如图 3 所示。主要初始

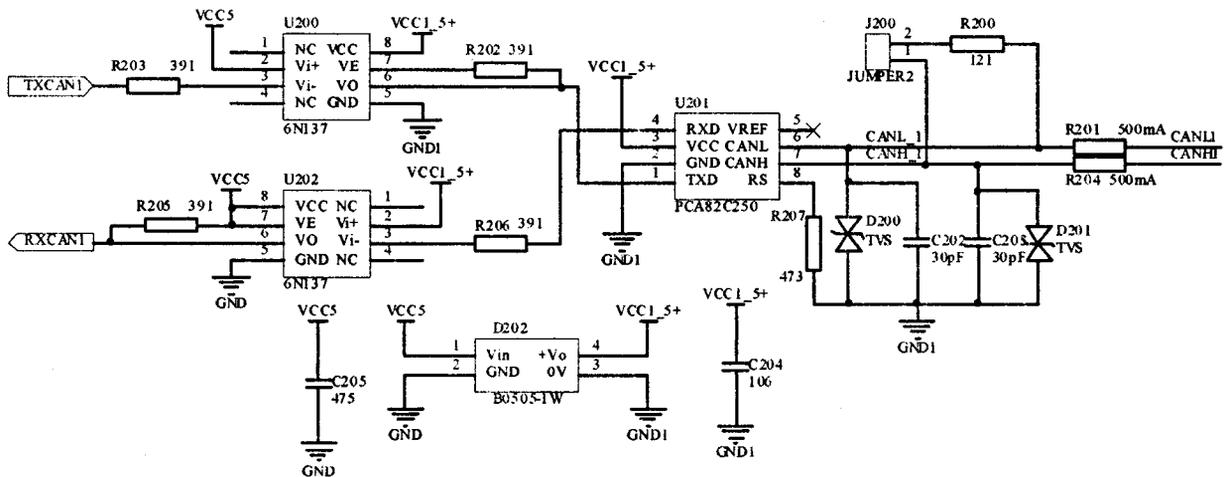


图 2 CAN 总线接口电路

化内容包括复位 CAN、设置总线波特率、设置 CAN 报警界限、设置中断工作方式、设置 CAN 控制器的工作模式、设置 CAN 验收过滤器等。经过以上初始化，CAN 模块就进入了所设置的工作模式了。

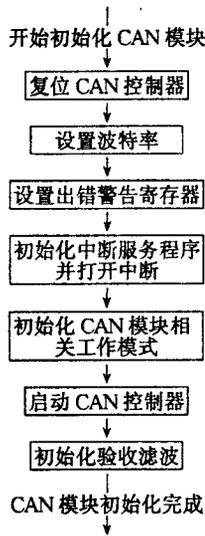


图 3 CAN 模块的初始化流程

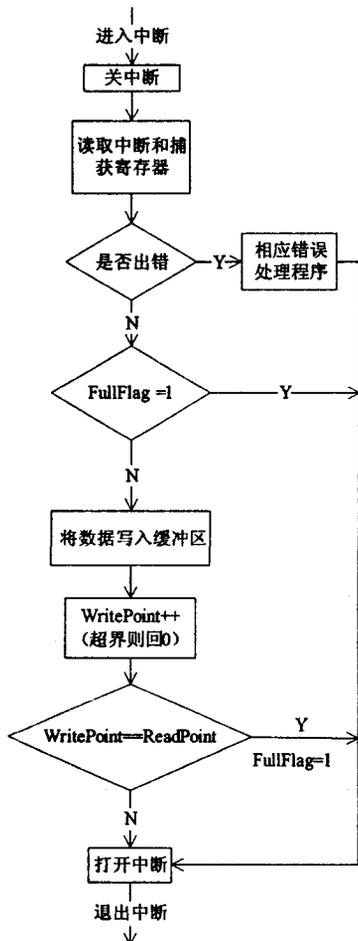


图 4 接收中断流程图

为保证系统可靠性，防止因为数据量大时发生数据溢出，在每个通道都引入了 FIFO 作为 CAN 总线接

收数据缓存。该 FIFO 是环形封闭的。用 WritePoint 来标记当前存储位置，用 ReadPoint 来标记当前读取位置，用 FullFlag 来标记缓冲区是否已满。当 WritePoint 不等于 ReadPoint 或 FullFlag 为 1 时，就表示缓冲区里面有数据等待发送，只有当 WritePoint 等于 ReadPoint 且 FullFlag 为 0 时，才表示缓冲区里没有有效数据。用中断服务来接收 CAN 总线上的数据，接收流程如图 4 所示。

数据发送用查询对应通道缓冲区标志 WritePoint, ReadPoint 和 FullFlag 来确定缓冲区里是否有数据等待发送。当 WritePoint 不等于 ReadPoint 或 FullFlag 为 1, 表示 FIFO 中有数据等待发。CAN 中继器数据发送流程图如图 5 所示。

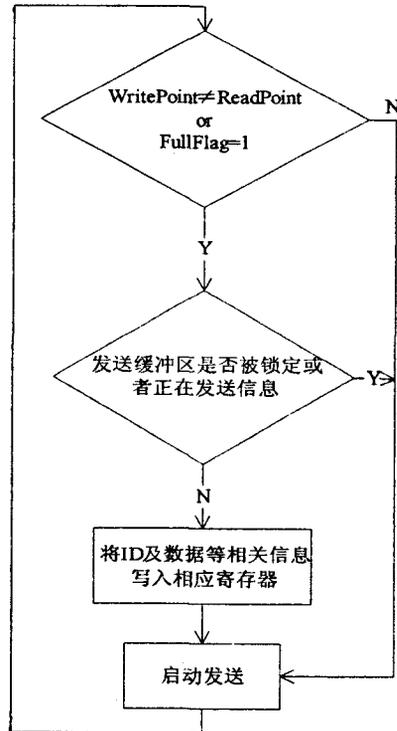
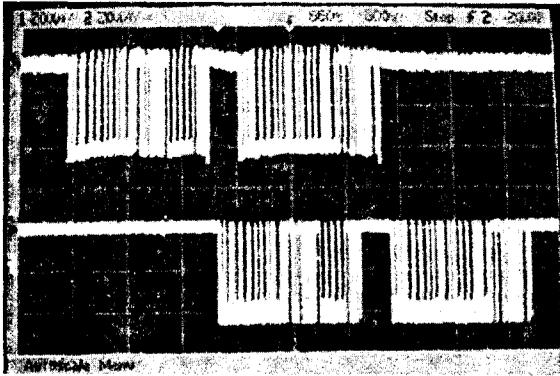


图 5 数据发送程序流程图

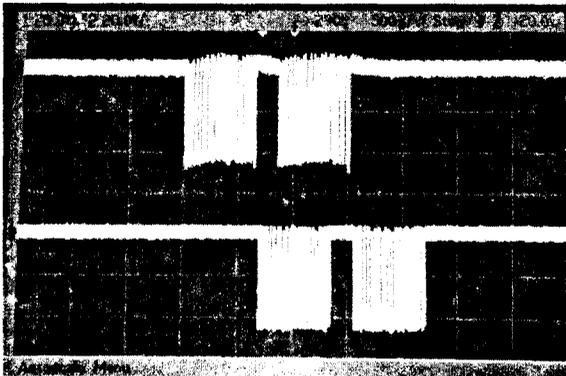
### 3 测试结果

将 CAN 中继器接入两个 CAN 总线网络之间，分别调整两个网络的波特率测试 CAN 总线中继器的性能。图 6(a)、(b)分别是两个 CAN 网络工作在 100kbps 和 200kbps 波特率下，CAN 中继器两个端口的波形图。

CAN 中继器中继过程是先存储，再转发，所以，有一定延时，而且，根据波特率不同，延时也不同，表 2 是 CAN 总线中继器在几个典型波特率下测得的延时。为了保证通信距离的要求，实时语音系统工作在 100kbps 的波特率下。1.32ms 的延时对该系统没有影响，没有丢包现象和语音明显延时现象。



(a) 100kbps



(b) 200kbps

图 6 相应波特率 CAN 总线中继器两个接口的波形图

### 4 结束语

CAN 总线中继器解决了增加网络节点和延长通信距离的问题,使得 CAN 总线网络扩展更简便容易,

也为两个不同波特率网络的相互通信提供了解决方案。该中继器在实际运行中可靠稳定,因为两个通道都设置了足够大的 FIFO,所以也没有出现数据溢出丢失的现象,传输延时也在可接受范围内。

表 2 在各种波特率下 CAN 中继器的时延

波特率	延时
50kbps	2600 $\mu$ s
100kbps	1320 $\mu$ s
200kbps	680 $\mu$ s
250kbps	550 $\mu$ s
500kbps	280 $\mu$ s

### 参考文献:

- [1] 饶运涛,邹继军,郑勇芸. 现场总线 CAN 原理与应用技术 [M]. 北京:北京航空航天大学出版社,2003:14-15.
- [2] 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,1996:25-32.
- [3] 王黎明,夏立. CAN 现场总线系统的设计与应用 [M]. 北京:电子工业出版社,2008:53-68.
- [4] 周立功. 深入浅出 ARM7——LPC213x\214x(上) [M]. 北京:北京航空航天大学出版社,2005:94-108.
- [5] PHILIPS Semiconductor. PCA82C250 CAN controller interface [S]. 2000.
- [6] 周立功. ARM 微控制器基础与实战 [M]. 北京:北京航空航天大学出版社,2005:218-256.
- [7] Liu Jianxin, Guan Xuefeng, Tan Ping. The analysis and test of real-time performance for time-triggered CAN bus [J]. Proceedings of the IEEE, 2008, 9(3):3005-3009.
- [8] PHILIPS Semiconductor. LPC2119/2129/2194/2292/2294 USER MANUAL [S]. 2004.

(上接第 197 页)

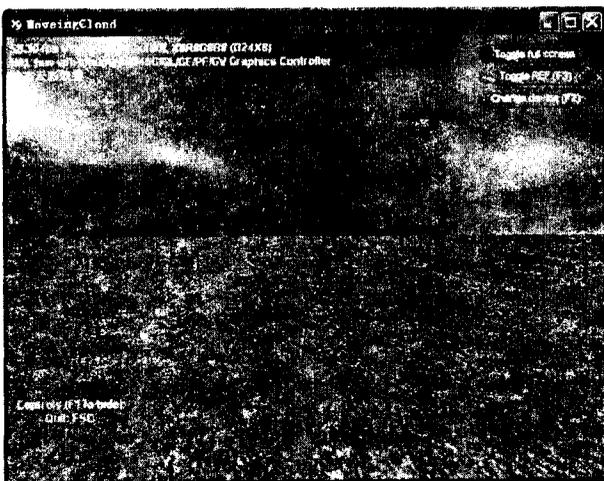


图 2 生成效果图

是动态的,但是如果纹理更新频率掌握不好,就会造成画面突变或者重复现象明显。这些问题都将在以后的工作中认真研究并加以解决。

### 参考文献:

- [1] 尹勇. 自然现象的实时仿真 [J]. 系统仿真学报, 2002, 14(9):1217-1219.
- [2] Max N. The Simulation of Natural Phenomena Panel [J]. Computer Graphics, 1983, 17(3):137-139.
- [3] 王德才,杨关胜,孙玉萍. 精通 DirectX 3D 图形与动画程序设计 [M]. 北京:人民邮电出版社,2007.
- [4] 陈卡. Direct 3D 图形程序设计 [M]. 上海:上海科学技术出版社,2003.
- [5] Sanchez J, Canton M. DirectX 3D 图形编程宝典 [M]. 韩传钊译. 北京:电子工业出版社,2000.
- [6] Crow F C. Summed-Area tables for texture mapping [J]. Computer Graphics, 1984, 18:207-212.
- [7] 侯健,李小奇,卢永吉,等. DirectX 中纹理映射技术的实现 [J]. 计算机技术与发展, 2008, 18(6):137-139.
- [8] 张芹,吴慧中,张健. 基于粒子系统的建模方法研究 [J]. 计算机科学, 2003(8):144-146.