

文章编号: 1006-2475(2014)02-0173-05

电动汽车 BMS 测试系统的数据通信

杨刘倩, 詹昌辉

(合肥国轩高科动力能源股份公司工程研究院, 安徽 合肥 230011)

摘要: 随着新能源汽车产业的发展, 车载动力电池管理系统(BMS)的可靠性是直接影响电动汽车可靠运营的关键因素之一。为此开发一套 BMS 测试系统, 实现对车载主从分布式 BMS 信息的监控及系统测试功能, 为 BMS 的测试及故障诊断提供便捷的途径。此测试系统是上位机与 BMS 通过 CAN 通信进行信息交互, 达到测试及监控的目的。本文重点描述 BMS 测试系统的上位机与 BMS 控制器 CAN 数据通信的实现过程及方法。

关键词: 电动汽车; BMS; 测试系统; CAN; 通信

中图分类号: U469.72 文献标识码: A doi: 10.3969/j.issn.1006-2475.2014.02.039

Data Communication in Testing System of Electric Car BMS

YANG Liu-qian, ZHAN Chang-hui

(Engineering Office, Hefei Guoxuan High-tech Power Energy Co., Ltd., Hefei 230011, China)

Abstract: With the development of new energy automotive industry, the reliability of vehicle battery management system (BMS) is one of the key factors to ensure electric cars work reliable. We develop a BMS testing system which provides convenient testing method for testing and fault diagnosis of BMS. It achieves the functions of monitoring and testing the distributed BMS. This test system achieves the testing and monitoring between the upper machine and BMS by CAN communication to exchange the information. This paper focuses on the implementation method of CAN data communication between the upper machine and the BMS controller.

Key words: electric car; BMS; test system; CAN; communication

0 引言

电动汽车行业的发展, 对各电动汽车组成部分性能要求不断提升^[1]。因此, 开发了 BMS (Battery Management System, 电池管理系统) 测试系统, 对 BMS 信息的采集精度、数据传输速率、准确性进行测试分析。

在 BMS 系统开发过程中, BMS 测试系统对 BMS 的流程控制及其故障响应进行验证; BMS 生产过程中, BMS 测试系统对 BMS 进行进一步测试^[2], 保障系统功能完善; BMS 售后阶段, BMS 测试系统可对 BMS 进行故障诊断和排查。

目前, 电动汽车领域多采用 CAN 总线实现各控制单元之间的通信^[3]。BMS 测试系统亦通过 CAN 通信实现与 BMS 的信息交互, 即可实现对 BMS 系统的测试、状态监控及数据分析等功能。

1 CAN 通信原理及通信协议

1.1 CAN 通信介绍

CAN (Controller Area Network, 控制局域网络) 属于现场总线的范畴, 现场总线被誉为自动化领域的计算机局域网^[4]。CAN 为分布式控制系统实现各节点之间实时、可靠的数据通信提供了强有力的技术支持, 应用于工业自动化设备、船舶、医疗设备等领域。

CAN 总线是德国 Bosch 公司从 20 世纪 80 年代初为解决现代汽车中众多的控制与测试仪器之间的数据交换而开发的一种串行数据通信协议, 它是一种多主总线, 通信介质可以是双绞线、同轴电缆或光纤, 通信速率可达 1 Mbps。当信号传输距离达 10 km 时, CAN 仍可提供高达 50 kbps 的数据传输速率^[5]。CAN 总线通信接口中集成了 CAN 协议的物

收稿日期: 2013-10-29

作者简介: 杨刘倩(1987-), 女, 安徽怀远人, 合肥国轩高科动力能源股份公司工程研究院助理工程师, 硕士, 研究方向: 电动汽车电池管理系统, 系统软件测试; 詹昌辉(1965-), 男, 安徽桐城人, 高级工程师, 硕士, 研究方向: 新能源汽车, 电动汽车电子电气系统集成技术及充电站工程技术。

理层和数据链路层功能,可完成对通信数据的成帧处理。

在 CAN 总线中有 4 种不同的帧类型^[6]:

(1) 数据帧(Data Frame):数据帧带有应用数据。

(2) 远程帧(Remote Frame):通过发送远程帧可以向网络请求数据,启动其他资源节点传送它们各自的数据。远程帧包含 6 个不同的位域:帧起始、仲裁域、控制域、CRC 域、应答域、帧结尾。仲裁域中的 RTR 位的隐性表示为远程帧。

(3) 错误帧(Error Frame):能够报告每个节点的出错,由 2 个不同的域组成。第一个域是不同站提供的错误标志的叠加,第二个域是错误界定符。

(4) 过载帧(Overload Frame):如果节点的接收尚未准备好就会传送过载帧,由 2 个不同的域组成。第一个域是过载标志,第二个域是过载界定符。

1.2 电池管理系统的 CAN 通信协议

CAN 协议是建立在 OSI 的 7 层开放互连参考模型基础之上的。但 CAN 协议只定义了模型的最下面 2 层:数据链路层和物理层,仅保证了节点间无差错的数据传输。CAN 的应用层协议必须由 CAN 用户自行定义,或采用一些国际组织制订的标准协议。

BMS 所采用的 CAN 通信协议中数据链路层的规定主要参考 CAN2.0B 和 SAE J1939 的相关规定。CAN 扩展帧有 29 位标识符,可自行规定其含义,定义 29 位 ID,如图 1 所示。

Priority 3-bit	Reserved 1-bit	Data Page 1-bit	PDU Format 8-bit	PDU Specific 8-bit	Source Address 8-bit
-------------------	-------------------	-----------------------	---------------------	-----------------------	-------------------------

图 1 扩展帧 ID 定义

(1) 优先级 Priority 为 3 位,可以有 8 个优先级;保留位(Reserved)值一般固定为 0;数据页(Data Page)值现固定为 0;8 位的 PDU Format 为报文的代码;8 位的 PDU Specific 为目标地址或组扩展;8 位的 Source Address 为发送此报文的源地址。

(2) 接入网络的每一个节点都有名称和地址,名称用于识别节点的功能和进行地址仲裁,地址用于节点的数据通信;每个节点至少有一种功能,可能会有多个节点具有相同的功能,也可能一个节点具有多个功能。

2 BMS 测试系统的 CAN 通信结构及原理

2.1 测试系统与 BMS 的连接结构

BMS 测试系统上位机与 BMS 的 CAN 通信硬件连接示意图如图 2 所示: BMS 测试系统上位机应用软件以 PC 机为载体,在 Windows XP 操作系统下开

发的应用软件生成的 EXE 文件运行在 PC 上^[7],形成一个上位机结构,该上位机可进行 CAN 通信,上位机通过 USB 接口与 CAN 通信传输模块连接,CAN 通信传输模块通过 CAN_H、CAN_L 连接电池管理系统的 CAN 通信接口。

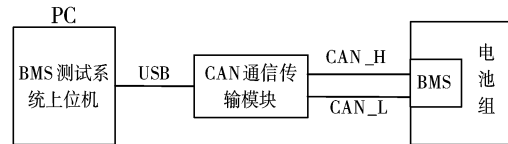


图 2 BMS 测试系统与 BMS 的 CAN 通信连接示意图

2.2 测试系统与 BMS 的 CAN 通信实现原理

测试系统应用程序在电脑上运行,电脑 USB 接口连接 CAN 通信数据传输模块,调用 CAN 通信的接口函数实现与下位机 BMS 的 CAN 通信连接,连接通信成功后,该应用程序可实现 CAN 报文的发送及接收功能。CAN 通信通过电平变化体现传输内容,CAN 通信传输模块将模拟信号转化成数字信号传递给 BMS 测试系统应用软件,软件接收信息后根据 CAN 通信协议解读、处理,用曲线图、图像、数字的形式将数据信息体现出来。

3 BMS 测试系统的软件流程及 CAN 通信实现

采用 Borland 公司开发的 Delphi7 作为 BMS 测试系统软件开发工具,完成软件界面设计与底层驱动的设计^[8],添加兼容 CAN 通信传输模块的 CAN 接口函数库,设计并开发具有 CAN 通信报文收发功能、解读 CAN 通信协议并对通信数据进行处理分析功能的应用软件^[9]。一方面,该应用软件将 CAN 报文信息转化成直观信息数据或图表在软件界面上显示,为单片机系统调试、测试工作提供便捷的途径。另一方面,BMS 测试系统的上位机应用软件可发送请求帧报文,BMS 接收到请求报文后回复所请求的数据或状态信息,实现信息交互。

3.1 通信流程的实现

首先设置好通信参数,根据下位机控制器 CAN 总线实际通信需要,将 CAN 总线通信速率设定为 250 kbps。将通信连接请求通过 CAN 数据传输模块传输至 BMS,通信连接成功后,向 BMS 发送相关测试信息请求报文或者直接接收来自 BMS 的广播信息报文,然后 BMS 测试系统应用程序对 CAN 报文接收缓存中的报文进行分析处理,将数据或状态信息显示或存储;对 BMS 系统信息进行实时监测时,CAN 通信是个循环过程,将循环发送请求或接收广播报文,当监测结束条件达到,退出该循环。最后完成 BMS 系统测试操作后关闭 CAN 通信连接^[10]。详细流程如图 3

所示。

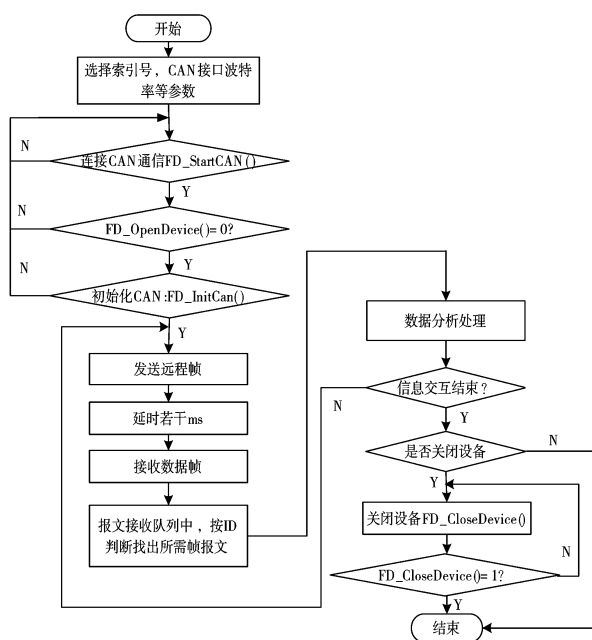


图3 CAN通信实现流程

3.2 CAN通信主要流程及示例程序

3.2.1 设定CAN通信传输参数 建立CAN通信连接

CAN通信硬件连接如图2所示,采用CAN通信数据传输模块连接BMS测试系统和BMS,CAN通信协议采用SAE J1939,建立BMS测试系统工程,首先设计图4所示窗体及底层驱动^[11-12],用来输入参数,建立通信连接。定义CAN通信时所需用到的结构体及函数体,在Delphi7建立的应用软件工程中调用该相关函数来实现串口通信。

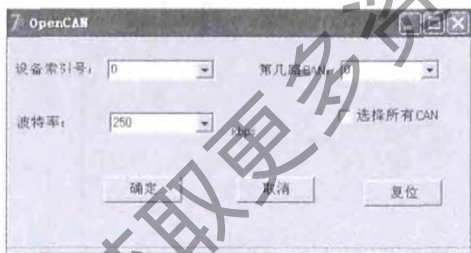


图4 BMS测试系统中CAN通信连接窗体

CAN通信数据模块通信连接结构体定义如下:

```
Type TVCI_INIT_CONFIG = record
//定义 VCI_INIT_CONFIG 结构体
AccCode: dword; //验收码
AccMask: dword; //屏蔽码
Reserved: dword; //保留位
Filter: BYTE; //滤波方式
BaudRate: integer; //波特率
Mode: BYTE //模式
end;
```

本文的CAN通信连接实现调用了CAN数据传输模块的接口函数库中函数FD_StartCAN,如:

```
function FD_StartCAN( DevType: dword; DevIndex: dword;
CANIndex: dword) : dword; stdcall;
```

3.2.2 发送远程帧请求信息报文

(1) 在Delphi工程中定义一个帧结构体,用来传输CAN信息帧,如:

```
Type TVCI_CAN_OBJ = record
//定义 VCI_CAN_OBJ 结构体,被用来传送CAN信息帧
ID: integer; //报文ID
TimeStamp: integer; //接收到信息帧时时间标
TimeFlag: byte; //是否使用时间标识
SendType: BYTE; //发送帧类型
RemoteFlag: BYTE; //是否是远程帧
ExternFlag: BYTE; //是否是扩展帧
DataLen: BYTE; //数据长度
Data: array[0..7] of Byte; //报文的数据
Reserved: array[0..2] of Byte; //系统保留
end;
```

(2) 定义一个过程array,用来初始化即将要发送的帧结构体,它的形参只有报文的ID,其他信息都赋给默认值。这样,在需要通过读取某一个报文信息时,只需传递相应的ID,发送一个远程帧,请求相应的报文信息,如:

```
procedure TForm1.array( const num: integer);
//初始化发送帧结构体
begin
arrsend0.ID:= num;
//arrsend0是上面定义的TVCI_CAN_OBJ记录类型的变量
arrsend0.TimeStamp:= 0;
arrsend0.TimeFlag:= 1;
arrsend0.SendType:= 0;
arrsend0.ExternFlag:= 1;
arrsend0.RemoteFlag:= 1;
arrsend0.DataLen:= 0;
arrsend0.Data[0]:= 0;
arrsend0.Data[1]:= 0;
...
arrsend0.Data[7]:= 0;
end;
```

(3) 调用函数FD_Transmit:

```
Function FD_Transmit( DevType: dword; DevIndex: dword;
CANIndex: dword; psend: pVCI_CAN_OBJ; len: integer) : integer; stdcall;
```

DevType是设备类型,DevIndex是设备索引号,CANIndex表示第几路CAN,psend是要发送数据帧数组的首指针,len是要发送的数据帧数组的长度。

3.2.3 接收数据帧报文

调用函数FD_Receive接收单片机控制器发过来的信息报文:

```
function FD_Receive( DevType: dword; DevIndex: dword;
CANIndex: dword; preceive: pVCI_CAN_OBJ; len: integer; wait-
```

time: integer): integer; stdcall;

DevType 是设备类型, DevIndex 是设备索引号, CANindex 表示第几路 CAN, preceive 是要接收的数据帧数组的首指针, len 是要发送的数据帧数组的长度, waittime 是等待时间。

另外, 需定义一个动态数组 list, 将接收到的数据放到动态数组里。

3.2.4 对接收到报文信息分析处理

在实际工程中, 上位机应用软件通过 CAN 数据模块对接收到的报文信息分析处理后显示在界面上, 或将信息存档。

(1) 接收的一条数据帧报文信息存入数组 list 中, 每个数据帧可传输 8 字节数据, list [0] 用来存放报文的 ID, list [1] 至 list [8] 存放数据内容。通过判断 list [0] 的内容, 对应通信协议分辨出该报文用来传输的数据代表信息内容。

(2) 对数组中数据进行转换, 得到具体参数值或系统状态。

(3) 通过显示控件在软件界面上显示出 BMS 的信息参数或状态。

3.3 CAN 通信数据实时数据处理

(1) 通信数据存储和读出。

利用控件 OpenFileDialog 和 SaveDialog 实现文件打开、保存功能, 对应用软件底层驱动代码编写实现 BMS 测试系统读取的通信信息的存储和读出^[13]。保存信息操作代码如下:

```
if savedialog2. Execute then
Begin
```

```
savedialog2. filename: = ChangeFileExt ( SaveDialog2.
FileName, '. ccm');
dbfname: = savedialog2. FileName;
AssignFile( F, DBFname); //初始化一个文件变量 F;
Rewrite( F); //能创建一个新文件并打开它;
try
For i: = Low( Members) to High( Members) do
Write( F, Members [ i ] );
finally
CloseFile( F); //关闭文件;
end;
```

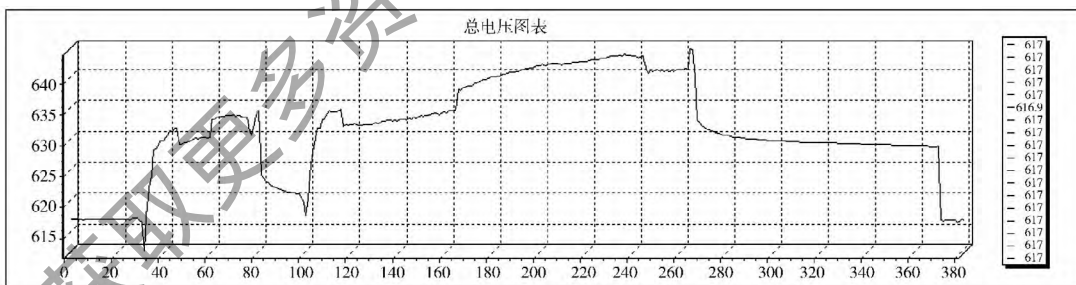
(2) 实时绘制参数动态曲线图^[14]。

在窗体中加入控件 chart 及 Timer, 通过 Timer 控件设定时间间隔触发, 每间隔 500 ms 更新数据绘制到 chart。以时间为横轴, 参数值为纵坐标。其中, 坐标点绘制采用函数 Addxy(X, Y)。

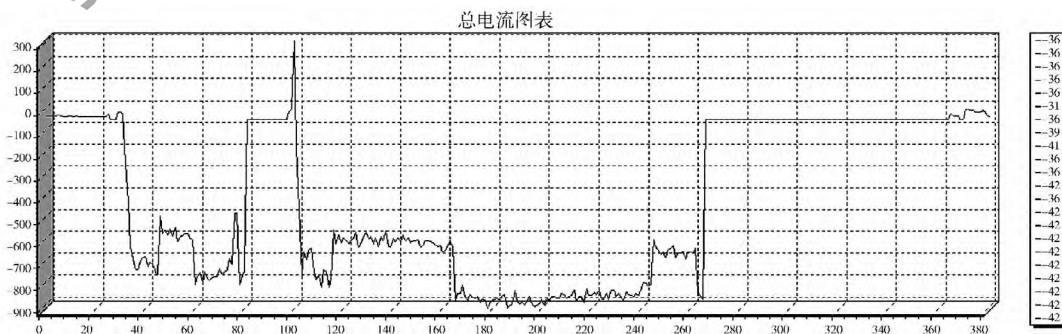
通过 CAN 通信, 上位机与下位机单片机控制系统进行信息交互, 得到下位机中系统参数后, 立即对实时数据进行直观显示, 并体现出数据变化趋势, 因此选用 chart 控件绘图。

4 实验结果与分析

BMS 测试系统可监控 BMS 各种数据信息, 观测 BMS 是否正常运行, 如监控电池组总电压及电池组总电流数据信息。BMS 测试系统以读取的总电压或总电流数值为纵坐标, 间隔 500 ms 绘制一个坐标点。随着监控时间、电压值及电流值的变化, 曲线动态地变化, 如图 5 所示。



(a) 电池组总电压曲线图



(b) 电池组总电流曲线图

图5 CAN 通信实时数据曲线图

监控信息用来分析 BMS 运行状态,为系统反馈提供依据。图 5(b) 中电池组电流值为负,相应图 5(a) 中电池组电压值处于增大趋势,表示 BMS 处于充电控制状态, BMS 测试系统给出电池组充电状态的指示。

5 结束语

本文描述用 Delphi7 开发设计 BMS 测试系统上位机应用软件,实现与电动汽车的 BMS 系统的 CAN 通信的功能,通过 CAN 通信完成对 BMS 的测试、监控,并对参数及状态做出分析、信息存储等。这一功能的实现,为工控生产、售后服务提供一个跳过通信协议分析过程的智能测试、监控的工具。另外,本 BMS 测试系统上位机应用软件界面友好,易操作,可移植,具有很好的实用性。

参考文献:

- [1] 袁方伟,陈思忠. 电动汽车电池管理系统的研究[J]. 汽车研究与开发,2003(3):41-44.
- [2] 张美云,潘玲. 国内外电动汽车的发展概况和趋势[J]. 广西机械,1997(3):19-22,13.
- [3] 曾光群. 基于无线通信技术的锂电池监测系统的设计与仿真研究[D]. 重庆:重庆大学,2007.

- [4] 吴慧英. 基于 CAN 总线的工业测控系统的应用研究[D]. 北京:北京航空航天大学,1996.
- [5] 罗峰,孙泽昌. 汽车 CAN 总线系统原理、设计与应用[M]. 北京:电子工业出版社,2010:31-35.
- [6] 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,1996:20-23.
- [7] 黄芳. Delphi7 存取配置文件实现与 SQL Sever 数据库连接[J]. 电脑编程技巧与维护,2012(10):44-45,77.
- [8] 赵晨,吴锤红. Delphi 高级通信调试助手软件的设计[J]. 福建农机,2007(3):33-39.
- [9] 孔令彬,叶敦范,付闯. 微机控制蓄电池监测系统[J]. 微电子技术,2001,29(6):35-38.
- [10] 张敬华. Delphi5 开发与应用[M]. 北京:北京航空航天大学出版社,2000.
- [11] 徐谔,张雷,徐立,等. Delphi 应用与开发案例教程[M]. 北京:清华大学出版社,2005:57-67.
- [12] 明日科技,梁水,李方超,等. Delphi 开发技术大全[M]. 北京:人民邮电出版社,2007:539-545.
- [13] [美]Marco Cantu,等. Delphi7 从入门到精通[M]. 罗征,等译. 北京:电子工业出版社,2003:39-63.
- [14] 高德欣,尚继良,王景景,等. Delphi 编程中的串口通讯技术[J]. 青岛科技大学学报:自然科学版,2003,24(21):125-127.

(上接第 114 页)

- [2] 程国雄,胡世清. 基于 Silverlight 的 RIA 系统架构与设计模式研究[J]. 计算机工程与设计,2010,31(8):1706-1709,1713.
- [3] 陈荔,张菁. 基于串口自定义协议传输文件的设计[J]. 计算机技术与发展,2012,22(2):45-48.
- [4] Dino Esposito, Andrea Saltarello. Microsoft .NET: Architecting Applications for the Enterprise [M]. Microsoft Press, 2010.
- [5] Laurent Bugnion. MVVM Light Toolkit [EB/OL]. <http://www.galasoft.ch/mvvm>, 2012-12-10.
- [6] Jarnjak F. Flexible GUI in robotics applications using Windows presentation foundation framework and model view viewmodel pattern [C]// Proceedings of the 4th International Conference on New Trends in Information Science and Service Science. 2010:176-179.
- [7] Gao Beibei, Zhang Shudong, Yao Naiming. A multidimensional pivot table model based on MVVM pattern for rich Internet application [C]// Proceedings of the 2012 International Symposium on Computer, Consumer and Control.

- [8] 程国雄,胡世清. 数字化图书及英语交互学习 RIA 平台的研究[J]. 计算机工程与设计,2010,31(13):3103-3107.
- [9] Josh Smith. WPF Apps with the Model-View-ViewModel Design Pattern [EB/OL]. <http://msdn.microsoft.com/en-us/magazine/dd419663.aspx>, 2009-02-28.
- [10] John Papa. Data Binding in WPF [EB/OL]. <http://msdn.microsoft.com/en-us/magazine/cc163299.aspx>, 2007-12-05.
- [11] [美]Matthew MacDonald. WPF 编程宝典: C# 2010 版 [M]. 王德才译. 北京:清华大学出版社,2011.
- [12] 傅棋灿,史浩山. 基于 Linq to NHibernate 数据库应用系统的设计与实现[J]. 微型电脑应用,2010,26(12):44-46.
- [13] Google. Google Maps JavaScript API V3 Reference [EB/OL]. <https://developers.google.com/maps/documentation/javascript/reference?hl=zh-CN>, 2013-05-30.
- [14] Dino Esposito, Andrea Saltarello. Microsoft .NET 企业级应用架构设计 [M]. 北京:人民邮电出版社,2010.