

# 基于 VC++ 串口通信软件的设计与实现

贾富强<sup>1</sup>, 赵 阳<sup>2</sup>

(1. 兰州交通大学 光电技术与智能控制教育部重点实验室, 甘肃 兰州 730070;  
2. 中铁工程设计咨询集团有限公司济南设计院, 山东 济南 250000)

**摘 要:** 由于在对有轨电车道岔和信号机驱动板的通信调试过程中, 存在上位机未设计好或者故障等因素, 从而造成对联调进度的影响。针对这种情况, 设计一种基于 VC++ 串口通信的有轨电车道岔和信号机驱动板通信测试软件, 直接通过测试软件代替上位机对驱动板进行通信调试, 以提高调试效率。测试结果表明, 该测试软件能够替代上位机的通信功能, 完成对驱动板的通信调试, 很大程度上提高了整个系统的联调效率, 并且证明该软件简单易用, 稳定可靠。

**关键词:** 串口通信; 道岔; 信号机; CSerialPort 类; 多线程

**中图分类号:** TP311.1

**文献标识码:** A

**文章编号:** 1673-629X(2015)06-0158-05

**doi:** 10.3969/j.issn.1673-629X.2015.06.035

## Design and Implementation of Serial Communication Software Based on VC++

JIA Fu-qiang<sup>1</sup>, ZHAO Yang<sup>2</sup>

(1. Key Laboratory of Opto-technology and Intelligent Control of Ministry of Education,  
Lanzhou Jiaotong University, Lanzhou 730070, China;

2. Jinan Institute of China Railway Engineering Consulting Group Co., Ltd., Jinan 250000, China)

**Abstract:** During the driver board communication debugging process of tram turnout and signal, there are factors effecting joint testing schedule, such as upper computer design unfinished or fault. For this case, a driver board of tram turnout and signal communication testing software based on VC++ serial communication is designed, which can substitute upper computer to debug the communication with driver board, improving debugging efficiency. The test results show that the software can achieve communication function of upper computer and complete the driver board communication debugging. It improves debugging efficiency of whole system, and has been proved that the software is easy and available, stable and reliable.

**Key words:** serial communication; turnout; signal; CSerialPort class; multi-threaded

## 0 引言

随着计算机技术和现代控制理论的快速发展, 各类控制系统的规模越来越大, 系统功能设计与实现趋于模块化, 因此系统中各个功能模块之间能否实现可靠通信成了整个系统协调而稳定工作的关键。串行通信是各类通信方式中最基本、最常用、最可靠的一种, 具有实现简单、使用灵活、数据传输可靠等优点, 因此采用串口通信实现计算机与其他设备的通信在系统检测、数据采集和实时监控等系统中的应用越来越广泛<sup>[1]</sup>。VC++是面向对象的可视化开发工具, 具有简便易用、功能强大、代码执行速度快等特点, 在通信软

件的开发中成为越来越多开发人员的首选工具<sup>[2]</sup>。文中采用 VC++ 开发软件, 利用多线程串口编程工具 CSerialPort 类进行串口通信设计与实现, 开发一种通信测试软件, 主要测试有轨电车道岔和信号机驱动板的通信功能。测试结果表明, 数据交换正常, 系统运行稳定可靠。除此之外, 该软件还可以作为串口调试助手使用。

## 1 系统硬件及其通信协议

### 1.1 通信硬件连接

串行接口是计算机与外部设备之间进行数据传输

的一种接口(简称串口)。美国电子工业协会(EIA) 订制并发布的串行数据接口标准主要有 RS-232, RS-422 和 RS-485。其中,RS-232 被定义为在低速串行通信中增加通信距离的单端标准,即单端通信标准,被广泛应用于计算机串行接口外设连接。由于其传输的

最大距离约为 15 m,最高传输速率为 20 kb/s,所以 RS-232 适合本地设备间的通信<sup>[3]</sup>。本系统中,由于道岔驱动板和信号驱动板均采用 CAN 通信,因此,测试系统与驱动板连接时,需要串口/CAN 转换器,其硬件连接如图 1 所示。



图 1 系统与道岔/信号机驱动板硬件连接图

1.2 通信协议

通信协议是通信双方为了完成通信或服务所必须遵循的规则和约定<sup>[4]</sup>。该系统中所编制的通信协议主要有测试软件与道岔驱动板间通信协议和测试软件与信号机驱动板间通信协议。系统要求 PC 机输出格式为串行异步全双工 RS-232 电气接口标准,要求测试软件每隔 250 ms 定时发送一次数据,并且同时发送和接收正码和反码两帧数据,正反码帧数据间延时 20 ms,通信序号在 0X00-0XFF 范围内连续递增。由于在高速率的数据传输过程中可能存在数据丢失现象,为了保证数据传输的正确性,需要对发送和接收的数据进行校验。系统采用 CRC 检验,对串口发送和接收的帧数据的 8 byte 数据位通过 CRC 校验,计算出的校验值存储在数据位后的两个字节中,其中前一个字节中

存高 8 位,后一个字节中存低 8 位。如果接收的帧数据正确,则进行数据解析处理,否则,丢弃帧数据。

(1)测试软件与道岔驱动板间通信协议。

测试软件与道岔驱动板间通信协议主要有测试软件向道岔驱动板下发控制命令帧数据通信协议和道岔驱动板向测试软件上传状态帧数据通信协议。测试软件与道岔驱动板间的通信帧的格式要求包含 12 位,帧格式如图 2 和图 3 所示。

(2)测试软件与信号机驱动板间通信协议。

测试软件与信号机驱动板间通信协议主要有测试软件向信号机驱动板下发控制命令帧数据通信协议和信号机驱动板向测试软件上传状态帧数据通信协议。测试软件与信号机驱动板间的通信帧的格式要求包含 12 位,帧格式如图 4 和图 5 所示。

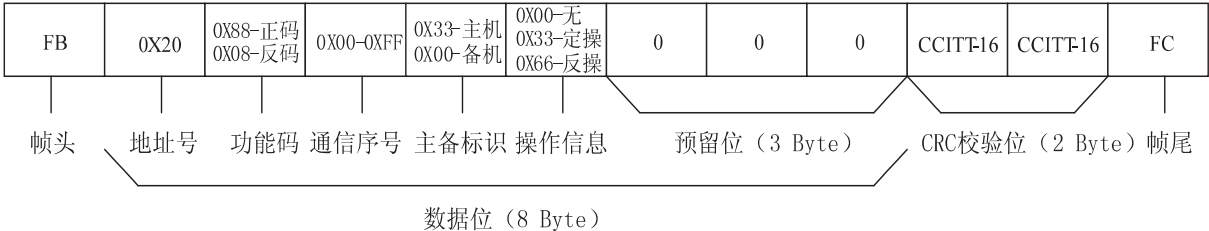


图 2 测试软件→道岔驱动板控制命令通信帧格式

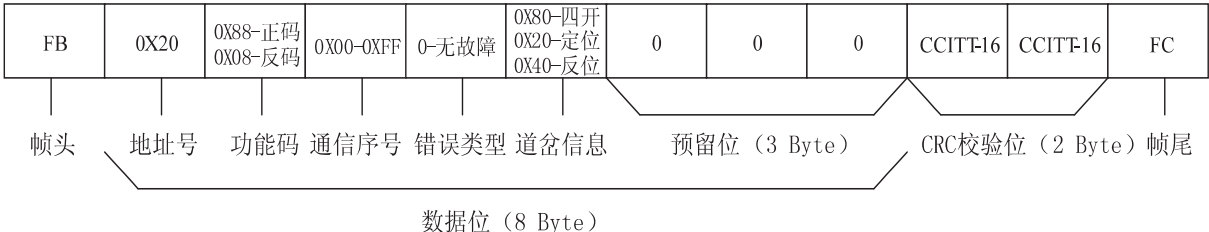


图 3 道岔驱动板→测试软件状态命令通信帧格式

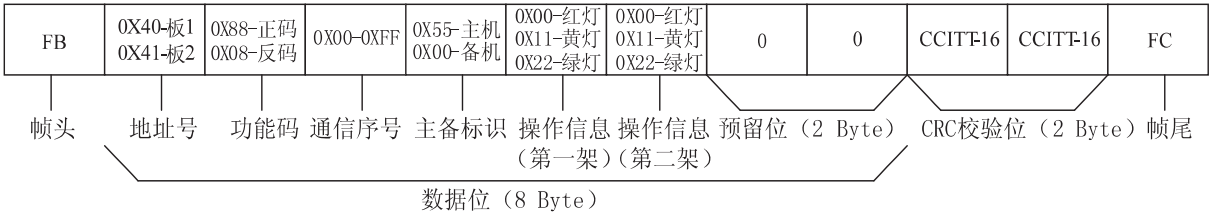


图 4 测试软件→信号机驱动板控制命令通信帧格式

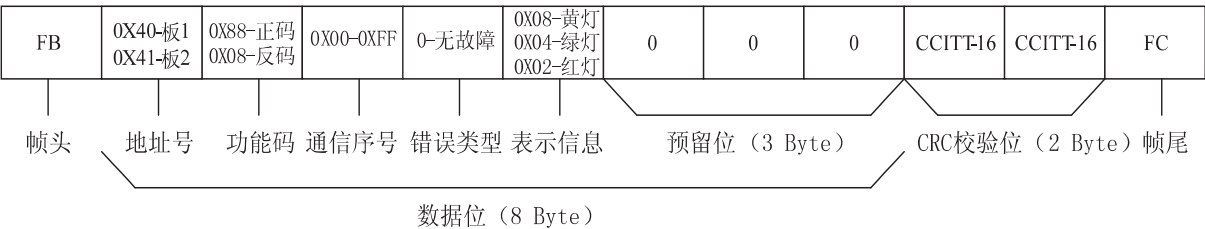


图 5 信号机驱动板→测试软件状态命令通信帧格式

2 软件设计与实现

2.1 CSerialPort 类

CSerialPort 类是由 Remon Spiekrijse 提供的免费串口类,它可以轻松地完成一般串口编程任务,几分钟就可以搭建好串口通信框架。CSerialPort 类支持线连接(非 MODEM)的串口编程操作,编写的程序在 Windows98/NT/2000/XP 操作系统下可很好地运行<sup>[5]</sup>。

CSerialPort 类是基于多线程的,其工作流程如下:首先设置好串口参数,再开启串口监测工作线程,串口监测线程监测到串口接收到的数据、流控制事件或其他串口事件后,就以消息方式通知主程序,激发消息处理函数来进行数据处理,这是对接收数据而言的;发送数据可直接向串口发送。

2.2 界面设计

打开 VC++ 6.0 环境下,通过 MFC AppWizard (exe)新建名为 TramSwicherTest 的单文档(SDI)工程,新建工程时所有选项均按默认设定。工程建好后,首先把单文档视图分割成四个视图窗口区,分别为道岔和信号机驱动板通信测试操作窗口区,道岔和信号机状态显示窗口区,数据发送窗口区和数据接收窗口区。并为每一个视图窗口区创建对应的视图类。分别为基于 CView 类的 CTramSwicherTestStationView 类,基于 CFormView 类的 CTramSwicherTestOrderView 类,CTramSwicherTestSendView 类和 CTramSwicherTestRecvView 类。由于基于 CFormView 的视图类需要有对话框与之对应,故应先在资源中新建与视图类功能相对应的测试操作对话框、数据发送对话框和数据接收对话框<sup>[6]</sup>。然后在三个对话框上根据设计需要添加所需控件,在状态显示窗口区绘制道岔和信号机。界面设计好后,通过程序实现各个视图区间的通信功能。

2.3 串口通信功能实现

根据该系统的通信协议要求,需完成通信串口的设置、数据发送、接收以及对收发数据的校验和对接收数据的解析。设计软件流程图如图 6 所示。

(1)添加 CSerialPort 类。

测试软件界面设计好后,将 CSerialPort 类添加到所建工程中<sup>[7]</sup>。首先将 CSerialPort.h 和 CSerialPort.cpp 复制到工程所在文件夹中,然后单击 VC 6.0 菜单

Project→Add to Project→Files,再在打开的文件选择对话框中选择 CSerialPort.h 和 CSerialPort.cpp,点击 OK,就把类文件加入到了所建工程中,并在 CTramSwicherTestOrderView.h 中添加头文件 CSerialPort.h 的声明:#include "SerialPort.h"。通过以上步骤就在当前工程中加入了 CSerialPort 类<sup>[8]</sup>。

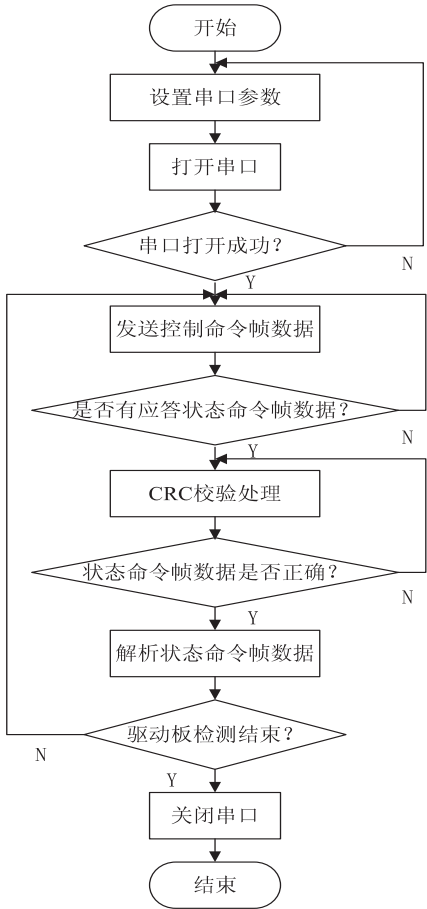


图 6 测试软件流程图

(2)初始化并打开串口。

程序中打开串口有两种方法:一是在单文档视图类的 OnInitialUpdate() 函数中添加串口打开程序,当程序启动时,调用单文档视图类中的 OnInitialUpdate() 函数,就可以打开串口;另外,通过单击设置好的“打开串口”按钮控件打开串口,由于检测软件运行时,需要进行串口参数的配置,所以通过设置好的“打开串口”按钮控件打开串口<sup>[9]</sup>。初始化并打开串口关键代码如下:

```
CSerialPort m_SerialPort; //CSerialPort 类对象
```

```

    BOOL m_bOpenPort;    //标志串口是否打开
    //以下为初始化串口(串口号:1,波特率:9 600 bit/s,无奇
    偶校验,8 数据位,1 停止位)
    m_nBaud=9 600;
    m_nCom=1;
    m_cParity='N';
    m_nDatabits=8;
    m_nStopbits=1;
    //以下为串口打开
    if(m_bOpenPort == TRUE)
    {
        m_ctrlOpenPort. SetWindowText(" 关闭串口");/*“打开串
        口”按钮文本显示
        if(m_SerialPort. InitPort( this, m_nCom, m_nBaud, m_cParity,
        m_nDatabits, m_nStopbits, EV_RXFLAG| EV_RXCHAR, 512))
        {
            m_SerialPort. StartMonitoring(); //启动监测辅助线程
            m_ctrlIconOpenoff. SetIcon( m_hIconRed); //串口打开图标显
            示
            SetTimer( 1, 250, NULL); //发送数据定时器
        }
        else
        {
            AfxMessageBox(" 没有发现此串口或被占用");
            m_ctrlIconOpenoff. SetIcon( m_hIconOff);
        }
    }

```

### (3) 发送数据。

检测软件通过单击界面上不同功能按钮实现向道岔驱动板和信号机驱动板发送相应的控制命令帧数据,根据通信协议要求,需要每隔 250 ms 发送一次数据。因此,需要设计一个定时器来控制数据的发送,定时器为 SetTimer(1, 250, NULL),并且要求同时发送正码和反码两帧数据,正码功能码为 0X88,反码功能码为 0X08,其他数据都相同,正码帧数据和反码帧数据间要求延时 20 ms,因此,在正码帧数据和反码帧数据发送程序之间添加 Sleep(20) 来实现延时。在这里使用 m\_SerialPort. WriteToPort(char \* string, int n) 来完成数据发送,其中定义了写到串口的字符串和字符的个数。在 CSerialPort 类中,补充了其他的 WriteToPort 的函数以满足其他情况下的需要,在使用过程中可以根据需要选择合适的写串口函数,也可以根据自己的需要来扩充其他函数<sup>[10]</sup>。

单击测试界面上不同功能的按钮,需要向驱动板发送的命令帧数据格式一致,数据包中具体数据内容有所不同,因此,在 CTramSwicherTestOrderView 类中创建一个发送函数 SendDataFun(char m\_Order),将每次的测试命令作为参数传给发送函数。定义两个数组分别对发送的正码帧数据和反码帧数据进行打包存储,

数组定义如下所示:

```

    unsigned char sendBufZM[12] = {0};    //发送正码帧数据
    数组
    unsigned char sendBufFM[12] = {0};    //发送反码帧数据
    数组

```

根据通信协议要求对数据进行 CRC 校验和打包处理,调用 CRC 校验函数 cal\_ccitt16(unsigned char \* buffer, unsigned int len) 对正、反码帧数据的 8 byte 数据位分别进行校验,校验后将帧头、帧尾、数据位和校验位打包处理,通过调用 CSerialPort 类的成员函数 WriteToPort() 完成数据的发送<sup>[11]</sup>,CRC 校验处理和发送过程如下所示:

```

    cal_ccitt16( sendBufZM+1, 8); //CRC 校验正码帧数据的 8
    byte 数据位
    cal_ccitt16( sendBufFM+1, 8); //CRC 校验反码帧数据的 8
    byte 数据位
    char sendArrayZM[12] = {0};
    char sendArrayFM[12] = {0};
    memcpy( sendArrayZM, sendBufZM, 12);
    memcpy( sendArrayFM, sendBufFM, 12);
    m _ SerialPort. WriteToPort ( sendArrayZM, sizeof ( sendAr-
    rayZM));
    Sleep(20); //延时 20 ms
    m _ SerialPort. WriteToPort ( sendArrayFM, sizeof ( sendArray-
    FM));

```

### (4) 接收数据。

根据通信协议要求,需要对接收的状态命令帧数据进行校验处理,如果接收数据正确则进行数据解析。首先将接收的数据存储于字符型数组中 recvBuff[12] 中,调用 CRC 校验函数 cal\_ccitt16(unsigned char \* buffer, unsigned int len) 对接收数据的 8 byte 数据位进行校验,将通过 CRC 校验算法计算出的两字节的校验位与接收帧数据中的校验位进行比较,如果一致则进行解析处理,否则,丢弃数据。串口接收数据的处理过程都在串口字符接收消息 WM\_COMM\_RXCHAR 的响应函数中<sup>[12]</sup>。其添加过程如下:

在 CSerialPort 类中有多个串口事件可以响应,在一般串口编程中,只需要处理 WM\_COMM\_RXCHAR 消息就可以了,该类所有的消息均需要人工添加消息处理函数<sup>[13]</sup>。本程序中将状态命令帧数据处理函数名定义为 OnCommunication,首先在 CTramSwicherTestOrderView. h 中添加串口字符接收消息 WM\_COMM\_RXCHAR 的响应函数声明:

```

    // { AFX_MSG( CV2Dlg)
    afx_msg LONG OnCommunication ( WPARAM ch, LPARAM
    port);
    // } AFX_MSG

```

然后,在控制区视图资源文件 (CTramSwicherT-



estOrderView. cpp) 中进行 WM\_COMM\_RXCHAR 消息映射:

```
BEGIN _ MESSAGE _ MAP ( CTramSwicherTestOrderView, CFormView)
//{ } AFX_MSG_MAP( CTramSwicherTestOrderView)
ON_MESSAGE( WM_COMM_RXCHAR, OnCommunication)
//{ } AFX_MSG_MAP
END_MESSAGE_MAP()
```

接着,在 CTramSwicherTestOrderView. cpp 中加入函数 OnCommunication() 的实现代码<sup>[14]</sup>。

### 3 软件测试及其结果分析

软件程序测试过程中,首先在本机上利用虚拟串口和串口调试助手进行演示,检查串口能否正常工作,测试软件能否按要求发送控制命令帧数据以及接收状态命令帧数据。接着连接硬件,利用开发的测试软件对道岔驱动板和信号机驱动板进行通信测试。

在此,以测试信号机驱动板 1 上的第一架信号机点红灯为例进行说明。首先运行程序,配置串口参数并打开串口,单击信号机 1 测试栏中的红灯按钮,信号机驱动板 1 上的第一架信号机红灯指示灯(1H)点亮,同时测试软件上的数据接收框中显示信号机驱动板 1 上传的状态命令帧数据,并且界面上的第一架信号机同步亮红灯。

### 4 结束语

文中利用 VC++ 编程工具和多线程串口编程工具 CSerialPort 类开发了一种通信测试软件,对 CSerialPort 类在 VC++ 中实现串口通信的方法进行了探讨,主要实现了对有轨电车道岔和信号机驱动板通信功能的测试。通过实例表明,利用 CSerialPort 类实现串口通信简单、方便、灵活;另外,所开发的测试软件还可以作为

串口调试助手使用。测试结果表明,该软件简单易用,数据通信可靠稳定,具有一定的实用价值。

#### 参考文献:

- [1] 汪 滢,孙华军. 基于 VC++ 串口通信的设计与实现[J]. 现代电子技术,2011,34(14):19-20.
- [2] 龚建伟,熊光明. Visual C++/Turbo C 串口通信编程实践[M]. 北京:电子工业出版社,2004.
- [3] 李景峰,潘 恒,杨丽娜. Visual C++/Turbo C 串口通信技术详解[M]. 北京:机械工业出版社,2013.
- [4] 王 娇,姚竹亭. 基于 VC++6.0 的串口通信设计[J]. 电子测试,2009(11):47-50.
- [5] 王 坤,高 贇. 基于 VC++ 实现串口通信的方法[J]. 信息化研究,2010,36(10):52-54.
- [6] 赵田雨,王崇宇. 单文档窗口分割视图和普通视图的切换[J]. 科技信息,2007(25):79-81.
- [7] 陈艾春. 基于 VC++ 串口通信[J]. 黑龙江科技信息,2010(1):82-82.
- [8] 丁云霞,曾连荪. 基于 VC++6.0 的 RS-422 串口通信设计[J]. 信息技术,2010(9):73-75.
- [9] Kruglinski D J, Wingo S, Shepherd G. Programming Visual C++ 技术内幕[M]. 第 5 版. 北京:北京希望电子出版社,2002:254-271.
- [10] Lenecker R C. Visual C++ Bible[M]. 北京:电子工业出版社,1999.
- [11] Denver A. Serial communications in Win32[M]. USA: Microsoft,1995.
- [12] 赵晓辉,陈艳萍,张科英,等. 基于 VC++ 环境下串口通信程序的研究与实现[J]. 科技信息,2010(1):39-40.
- [13] 王 臻,陈永泰. VC++ 实现串口通信的应用程序设计[EB/OL]. 2007-06-06. <http://www.paper.edu.cn/>.
- [14] Wang Qi, Zhu Hao, Gao Yingke. Accelerating processor verification based on ESL model[M]. Beilin: Springer - Verlag, 2012:745-751.

(上接第 157 页)

- 详:出版者不详,2002:191-195.
- [3] 王新安,蒋安平,宋春殚. 数字系统测试[M]. 北京:电子工业出版社,2007.
  - [4] 王新安,吉利久. SOC 测试中的 BIST 的若干思考[J]. 微电子学与计算机,2003,20(10):41-44.
  - [5] 桂江华,钱黎明,申柏泉,等. SOC 中的 MBIST 设计[J]. 电子与封装,2011,11(1):26-28.
  - [6] Rajsuman R. Design and test of large embedded memories:an overview[J]. IEEE Design and Test of Computers,2001,18(3):16-27.
  - [7] Rajsuman R. System-on-a-chip: design and test[M]. [s. l.]: Artech House Inc,2000.
  - [8] 余晓文. SRAM 时序单元设计及验证[D]. 合肥:中国科学

技术大学,2004.

- [9] 姚 俊. 基于 BIST 的嵌入式存储器可测性设计算法研究[D]. 哈尔滨:哈尔滨工程大学,2007.
- [10] 方祥圣,曹先霞. 系统芯片 SOC 的 BIST 测试研究[J]. 安徽建筑工业学院学报:自然科学版,2006,14(3):59-61.
- [11] 徐 磊. 基于 SOC 架构的可测性设计方法学研究[D]. 北京:清华大学,2002.
- [12] 陈 凯. 基于嵌入式核测试复用的数字 SOC 测试技术研究[D]. 桂林:桂林电子科技大学,2006.
- [13] 刘建都. CMOS 存储单元的开路故障可测性设计[J]. 现代电子技术,1994,22(2):10-14.
- [14] 刘建都,张安堂. CMOS 存储器中地址译码器的开路故障及测试[J]. 空军工程大学学报:自然科学版,2000,1(2):78-81.

# 基于VC++串口通信软件的设计与实现

作者：[贾富强](#)，[赵阳](#)，[JIA Fu-qiang](#)，[ZHAO Yang](#)

作者单位：[贾富强, JIA Fu-qiang\(兰州交通大学 光电技术与智能控制教育部重点实验室, 甘肃 兰州, 730070\)](#)，[赵阳, ZHAO Yang\(中铁工程设计咨询集团有限公司济南设计院, 山东 济南, 250000\)](#)

刊名：[计算机技术与发展](#)

英文刊名：[Computer Technology and Development](#)

年，卷(期)：2015(6)

引用本文格式：[贾富强](#), [赵阳](#), [JIA Fu-qiang](#), [ZHAO Yang](#) [基于VC++串口通信软件的设计与实现](#)[期刊论文]-[计算机技术与发展](#) 2015(6)