

# 基于 DSP 的 TSF 控制器串行通信设计

宗波, 黄杰, 蒋晓春, 王轩

(中国电力科学研究院, 北京 100192)

**摘要:** 为了满足电力电子装置日益复杂的功能要求, DSP (数字信号处理器) 在其控制器中的应用日益普遍。介绍了利用 TMS320F28335 DSP 芯片的 SCI (串行通信接口) 模块实现适用于 RS-485 总线的 TSF (晶闸管投切滤波器) 控制器串行数据通信方案, 并使用 VC++ 开发了相应的上位机软件。该方案可应用于多种基于 DSP 的电力电子控制装置与上位机间的数据交换。在实际环境下运行表明该方案简单有效、移植方便, 具有较强的实用性。

**关键词:** 数字信号处理器; TMS320F28335; 串行通信接口; VC++; RS-485

## Design of the serial communication of TSF controller based on DSP

ZONG Bo, HUANG Jie, JIANG Xiao-chun, WANG Xuan

(China Electric Power Research Institute, Beijing 100192, China)

**Abstract:** In order to meet the increasingly requirements of power electronic devices' complex function, the application of DSP (digital signal processor) is increasingly common in controller system. This paper describes how to use the SCI (serial communication interface) module of TMS320F28335 DSP to implement the TSF (Thyristor Switched Filters) controller serial data communication on RS-485 bus, and develops the corresponding host computer software based on the VC++. The program can be used in data exchange between variety of DSP-based power electronic devices and the host computer. The real environment test proved that this method is simple, effective and portable convenience, has strong practicability.

**Key words:** DSP; TMS320F28335; serial communication interface; VC++; RS-485

## 0 引言

现代电力电子技术在交流调速、供电电源、电力系统输配电、电能质量控制等方面已经获得日益广泛的应用。晶闸管投切滤波器(TSF)是电力电子装置的一种,它不仅能够动态地补偿负荷所消耗的无功功率,还能吸收负荷产生的大量谐波电流<sup>[1]</sup>。为了满足电力电子控制器日益复杂的功能要求,数字信号处理器在其控制器中的应用日益普遍,同时也对控制系统的可靠性和实时性提出了更高的要求。在 TSF 工程应用中,数字信号处理器与上位机的串行数据交互直接关系到 TSF 开发、调试与实验的整个过程,需要设计一种简单、可靠、移植性好的串行通信方案。

本文给出的串行数据通信方案中,控制系统选用 TI 公司 C2000 系列 DSP 产品 TMS320F28335。同时,使用 VC++ 中 MSComm 控件编写上位机软件,实现 DSP 与上位机间的数据交换。该方案实现了基于 DSP 的晶闸管投切滤波器控制器与上位机间的实时数据交换,有效缩短了晶闸管投切滤波装置的开发与研制周期,同时降低了晶闸管投切滤波装置的实验与维护费用。

## 1 通信系统硬件构成

F28335 具有三个串行通信接口,每个串行通信

收稿日期: 2009-12-11

作者简介: 宗波(1981-),男,硕士,助理工程师,从事电力电子技术、数字化控制技术的研究。

接口 SCI 的接收器和发送器各具有 1 个 16 级深度的先入先出数据缓存器 (FIFO); 具有各自独立的使能位和中断位, 可以进行半双工或全双工操作<sup>[2]</sup>。通过 1 个 16 位的波特率选择寄存器, 数据传输的速度可以被编程为 65535 种不同的方式。串行通信接口的数据采用 NRZ (非返回零) 格式。NRZ 数据格式包括: 1 个起始位、1~8 个数据位、1 个奇/偶校验位或无奇/偶校验位、1~2 个停止位、1 个用于区分数据和地址的额外位。

PC 的串行通信接口一般采用 RS-232 总线, 该总线传输速率低, 传输距离近, 抗干扰能力弱, 很难保证晶闸管投切滤波器实际工程现场作业数据的精确性和较远距离传输的要求。因此, 方案中采用了平衡差分接收的 RS-485 总线, 使得抗干扰能力和传输距离等性能得到了很大提升, 满足了实际需要, 并且可以根据实际需求, 进一步组成分布式网络, 实现一主多从网络。

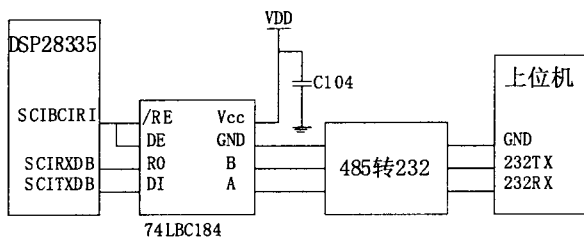


图 1 PC 与 F28335 串行通信接线图

图 1 所示为 PC 与 F28335 串行通信接线图, 其中 74LBC184 为 RS485 模块, 其接收器输出使能 /RE 和驱动器输出使能 DE 有效电平相反, 与 F28335 的一个通用口相连, 控制串口数据的输入和输出。

## 2 通信协议设计

在软件设计之前首先需要在异步串行通信的基础上开发出一个统一的协议。协议采用电力系统行业常用的 IEC60870-5-101<sup>[3]</sup> 规约的帧格式。方案中通信速率设定为 38400bps, 无校验位, 数据位 8 位, 停止位 1 位。

通信协议包含 2 种帧格式, 1 种是可变帧格式, 1 种是固定帧格式。可变帧格式可以用来实现主站 (上位机) 与从站 (TSF 控制器) 之间的数据传输。固定帧格式则可以用来实现主站与从站之间的查询和确认。帧格式如表 1-2 所示。

表 1 可变帧格式

停止位	D7	D6	D5	D4	D3	D2	D1	D0	起始位
									启动字符 (68H)
									数据长度 (L)
									数据长度 (L) 重复
									启动字符 (68H)
									控制域 (C)
									链路地址域 (A)
									链路用户数据 (可变量长度)
									帧校验和 (CS)
									结束字符 (16H)

表 2 固定帧格式

停止位	D7	D6	D5	D4	D3	D2	D1	D0	起始位
									启动字符 (10H)
									控制域 (C)
									链路地址域 (A)
									结束字符 (16H)

说明:

(1) 启动字符: 帧格式的特征码 68H 代表可变帧格式, 10H 代表固定帧格式。

(2) 数据长度: 包括控制域、地址域、用户数据区的 8 位位组的个数, 为二进制数。L = 1 字节 (控制域) + 1 字节 (控制域) + 用户数据个数 \* 2 字节

(3) 控制域: 包含传输方向位, 启动报文位, 帧计数位, 帧计数有效位以及主站到从站的功能码, 从站到主站的功能码等内容。

(4) 链路地址域: 地址域 (A) 的含义是当主站触发一次传输服务时, 主站向子站传送的帧中表示报文所要送达的目的站址, 即子站站址; 当由子站向主站传送帧时, 表示该报文发送的源站址, 即表示该子站站址。地址域的值 0 至 255, 其中 FFH = 255 为广播站地址, 即向所有站传送报文。这里规定上位机 (主站) 地址为 0x01, TSF 控制器 (从站地址) 为 0x02。

(5) 链路用户数据: 要传输的数据内容。这里每个数据占 2 个字节。

(6) 帧校验和: 帧校验和是控制域、地址域、用户数据区 8 位位组的算术和。

(7) 结束字符: 作为该帧数据的结束。

## 3 DSP 程序设计

### 3.1 DSP SCI 模块初始化

F28335 在通信之前要进行一系列的初始化, 包括系统初始化和 SCI 初始化等。其中波特率选择寄存器数值计算方法为:

$$BBR = (LSPCLK\_FREQ / (SCI\_FREQ * 8)) - 1$$

28335 系统时钟频率为 150MHz, LSP 频率为

150MHz/4,则 BBR 值为 121,换算为 16 进制即为 0x79。

SCI 模块宏定义如下:

```
#define CPU_FREQ          150E6
#define LSPCLK_FREQ      CPU_FREQ/4
#define SCI_FREQ         38400
#define SCI_PRD
(LSPCLK_FREQ/(SCI_FREQ * 8)) - 1
```

SCI 模块初始化函数完成以下功能:SCIB 模块的初始化,以及帧格式设定:1bit 停止位、无奇偶校验、8bits 数据位。另外完成空闲线唤醒使能,SCI 发送中断并使 SCI 退出复位。代码如下:

```
void scib_echoback_init()
{
    ScibRegs. SCICCR. all = 0x0007;
    ScibRegs. SCICTL1. all = 0x0003;
    ScibRegs. SCICTL2. all = 0x0003;
    ScibRegs. SCICTL2. bit. TXINTENA = 1;
    ScibRegs. SCICTL2. bit. RXBKINTENA = 1;
    ScibRegs. SCIHBAUD = 0x00;
    ScibRegs. SCILBAUD = 0x0079;
    ScibRegs. SCICTL1. all = 0x0023;
}
```

### 3.2 DSP 通信程序设计

TMS320F28335 串行通信可以采用查询和中断两种方式,其中查询方式是在查询到相应标志成立时,执行相应的动作(如发送一个字节)。这种工作方式要在串行口和接口电路间交换数据、状态和控制三种信息,致使 DSP 的利用率受到严重影响<sup>[4]</sup>。

方案中采用中断方式。DSP 启动串口后,不再询问其状态,继续执行主程序,直至串行口产生中断,DSP 响应后,开始执行相应的中断服务。

下面仅以接收数据为例,说明 DSP 芯片 SCI 模块如何工作。数据接收流程图如图 2 所示。

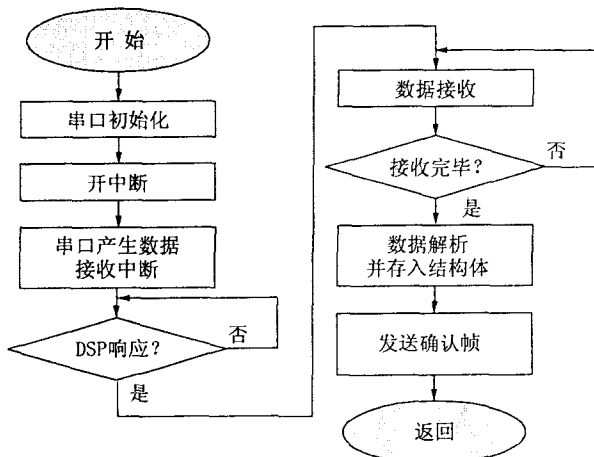


图 2 DSP 数据接收流程

当 SCI 模块接收数据后,首先判断接收数据是否完整。随后进行数据解析,解析时,先检测数据帧中的地址信号,如果和自身的地址不匹配,则丢弃该帧;否则进行相应的数据解析。解析数据时,首先根据帧头确认为变长帧或固定帧,随后进行帧尾确定,回报数据接收完毕。接下来进行校验和验证,如果错误,丢弃该帧数据;如果正确,则修改特征码,下发数据。当解析完成之后向上位机发出确认帧,返回主程序。

## 4 上位机程序设计

本上位机软件采用 VC ++ 作为上位机开发工具,通过使用 MSComm 控件实现快速,方便的设计目的。

MSComm 是 Microsoft 提供的 Windows 下串行通信编程的一个 ActiveX 控件,其核心内容是组件对象模型 COM,它以属性和事件的形式提供对 Windows 通信驱动程序 API 函数的接口。MSComm 控件使用事件驱动方式来处理和解决各类通信软件的开发设计,并提供了使用 RS232 进行数据通信的所有协议。VC ++ 为该控件提供了标准时间处理函数,并通过属性和方法提供对串行通信的设置。其中主要属性如下所示:

SetCommPort:指定使用的串口。

GetCommPort:得到当前使用的串口。

SetSettings:指定串口的参数。

GetSettings:取得串口参数。

SetPortOpen:打开或关闭串口,当一个程序打开串口时,另外的程序将无法使用该串口。

GetPortOpen:取得串口状态。

GetInBufferCount:输入缓冲区中接受到的字符数。

SetInPutLen:一次读取输入缓冲区的字符数。设置为 0 时,程序将读取缓冲区的全部字符。

GetInPut:读取输入缓冲区。

GetOutBufferCount:输出缓冲区中待发送的字符数。

SetOutPut:写入输出缓冲区。

SetRThreshold:设定引发 OnComm 事件的字节数。在设计接收可预知变长数据串时,在查询之前修改此属性将简化程序设计难度。

### 4.1 上位机程序设计思想

上位机用来对 TSF 上位机程序进行快速状态设置及查询。

上位机软件通过 RS232 接口与 DSP 控制器进行通信交互。上位机下发数据给 DSP 控制器时,DSP 控制器采用中断方式接收,DSP 控制器回报数

据给上位机时则采用查询方式。参数的下发采用先查询后下发的方式。查询完成后,修改要调整的参数再下发所有参数。

上位机软件首先完成必要的初始化,如操作窗口重绘、各变量的初始值设置,以及通讯串口的初始化操作等。

进入主循环后根据初始化数据(默认值)显示部分参数数据(必要定值),并自动查询当前控制器数据。在查询控制器数据结束之后,在主界面上对各参数进行显示,此时,用户可根据需要修改参数,并发送至控制器。再次查询之后,可以根据界面显示判断参数是否修改完成。

#### 4.2 上位机程序部分代码

在上位机软件代码实现中,要进行以下几个属性设置,包括通信端口号、通信协议、传输速率等,此外还需将所需要的串口打开<sup>[5]</sup>。

通讯串口 1 设置源代码:

```
BOOL CTSFSerialPortDlg::OnInitDialog()  
{  
    CDialog::OnInitDialog();  
    //此处为窗口初始化代码,在窗口创建完后,自动添加。  
    ...  
    //串口初始化代码。  
    m_mscom. SetCommPort(1); //使用串口 1  
    m_mscom. SetInBufferSize(1024); //设置输入缓冲区的大小,Bytes  
    m_mscom. SetOutBufferSize(512); //设置输出缓冲区的大小,Bytes  
    if(! m_mscom. GetPortOpen()) //打开串口  
    {  
        m_mscom. SetPortOpen(true);  
    }  
    m_mscom. SetInputMode(1); //设置输入方式为二进制方式  
    m_mscom. SetSettings("38400,n,8,1"); //设置波特率等参数  
    m_mscom. SetRThreshold(10); //为 10 表示有十个字符即引发事件  
    m_mscom. SetInputLen(0);  
    return TRUE;  
}
```

限于文章篇幅,将上位机接收数据代码节录如下:

```
void CTSFSerialPortDlg::OnOnCommMscomm()  
{
```

```
//串口事件响应函数  
VARIANT variant_inp;  
COleSafeArray safearray_inp;  
LONG len,k;  
BYTE rxdata[128]; //设置接受 BYTE 数组。  
CString strtemp;  
if(m_ctrlComm. GetCommEvent() == 2) //收到大于 RTHreshold 个字符  
{  
    variant_inp = m_ctrlComm. GetInput(); //读缓冲区  
    safearray_inp = variant_inp; //VARIANT 型变量转换为 COleSafeArray 型变量  
    len = safearray_inp. GetOneDimSize(); //得到有效数据长度  
    for(k=0;k<len;k++)  
        safearray_inp. GetElement(&k,rxdata+k); //转换为 BYTE 型数组  
    //此处调用解析函数进行数据处理  
    ...  
} ... //其他串口事件处理
```

在本程序设计过程中,涉及串口操作的函数全部进行模块化封装。在以后开发同类型的电力电子装置控制器时,只需要修改数据解析和数据封装函数,提高了代码复用率。

## 5 结束语

本文给出了一种基于 DSP 的 TSF 控制器与 PC 间的数据交换的设计方案,此方案通过 DSP 中的 SCI 模块,实现了控制器与 PC 间的串行数据通信,有效将各种系统参数控制在合适的范围之内,完成了电力电子装置开发调试过程中的定值参数传递与实验数据上传的功能。证明了该系统具有简单、可靠、开发周期短、移植性好等特点,有效缩短电力电子装置的开发与研制周期,对于基于 DSP 的电力电子装置控制器的数据串行通信有一定参考价值。

### 参考文献:

- [1] 王兆安,等. 谐波抑制和无功功率补偿[M]. 北京:机械工业出版社,2005.
- [2] TMS320F28335 Digital Signal Controllers(DSCs) Data Manual[Z]. Texas Instruments Incorporated, USA, 2007.
- [3] IEC 60870-5-101. Companion standard for basic telecontrol tasks[S].
- [4] 苏奎峰,等. TMS320X281X DSP 原理及 C 程序开发[M]. 北京:北京航空航天大学出版社,2008.
- [5] 胡志坤. Visual C++ 通信编程工程实例精解[M]. 北京:机械工业出版社,2007.

责任编辑:刘新影