

# GPRS DTU 数据中心的通信设计

## Design for Communication of Data Center of GPRS DTU

刘志雄 余臻 陈燕萍 (厦门大学信息科学与技术学院, 福建 厦门 361005)

### 摘要

介绍了基于 GPRS 网络的电力监控系统结构, 并根据 GPRS DTU 的特点, 结合网络编程的知识, 在 VC++6.0 平台上实现了 GPRS DTU 的数据中心的通信功能。

关键词: GPRS, 远程监控, 数据中心

### Abstract

This paper introduces the power monitoring and controlling system architecture, which is based on GPRS, and according to the traits of GPRS DTU, combining the knowledge of network programming. This paper implements the functions of communication on data centers of GPRS DTUs on the platform of VC++6.0.

Keywords: GPRS, Distant monitoring and controlling, data center

本文主要介绍了基于 GPRS 网络的电力监控系统结构, 并根据 GPRS DTU 与数据中心的通信特点, 介绍了数据中心与各个子站的 GPRS DTU 通信功能的设计与实现。

### 1 系统结构

组建一个基于 GPRS 网络的电力监控系统需要如下设备和程序:

1) GPRS DTU 设备和 SIM 卡。每个站点一般 1 台 DTU, 通过 RS232 或者 RS485 与子站的 RTU 相连, 实现与主站的数据传输。各个 DTU 需要一张 SIM 卡。卡向移动公司购买月租费比较便宜的卡, 根据实际数据流量需要, 申请开通合适的 GPRS 资费套餐。

2) 子站的电力参数采集设备和程序。该部分完成采集各种电力参数, 保存并根据主站的需要通过 GPRS 网络上传数据。

3) 数据中心服务器。可以采用普通的 PC 机, 该 PC 机要求连接上互联网, 有固定的 IP 地址。

4) 数据中心的监控系统程序。该系统可以实现与各个 DTU 通信, 并且具有监控系统的一般功能, 实现遥控、遥测、遥调、遥信“四遥”功能。

整个基于 GPRS 网络的电力监控系统结构如图 1。

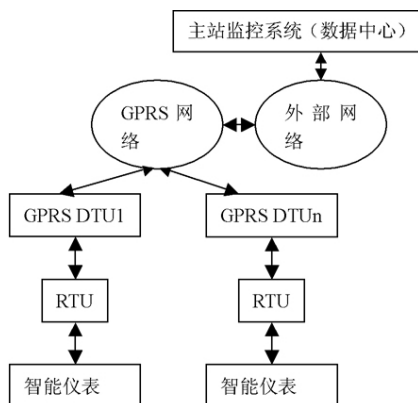


图 1 电力监控系统网络

### 2 GPRS DTU 特点与数据中心通信功能

GPRS DTU 的数据中心的通信功能, 主要是通过网络编程来实现的。在网络中, GPRS DTU 是客户端, 与普通的客户端相比, 有自己的特点, 需要对 GPRS DTU 的参数进行设置。在参

数中, 心跳包的时间设置值得认真考虑, 要考虑设置长多的时间; 此外还要注意 DTU 注册包消息, 因为其包含了对有数据中心很有用的消息。

#### 2.1 心跳包时间设置

GPRS 网络的特点是, 如果长时间没有数据通信, 那么移动网关将断开 GPRS DTU 与中心的连接。为了保持 GPRS DTU 永久在线, 典型的 GPRS DTU 在设计上包含上电自动拨号、采用心跳包、支持断线自动重连、自动重拨号这些功能, 使得 DTU 具有永久在线功能。采用心跳包保持永久在线时, 存在一个心跳包时间设置为多少比较合适的问题。如果心跳包时间太短了, 那么在 DTU 发送的心跳包由于网络延迟, 数据中心没有及时接收到心跳包, 那么 DTU 就会判断连接断开, 从而自动重连。相反, 心跳包时间设置太长, 而这时没有数据传输, 那么移动网关将断开 DTU 与数据中心的连接。通过大量的实验表明, 把心跳包时间设置为 15 分钟左右(厦门地区), 可以基本保证 DTU 永久在线。

#### 2.2 DTU 注册包

DTU 上电以后会根据设置的数据中心 IP 和端口号会主动去连接数据中心, 一旦 TCP 链路建立成功, DTU 将发送首次注册包, 其中包含了 DTU 的基本信息, 不同的厂家的 DTU 注册包应该不同。当 DTU 与数据中心在通信过程中, 发生异常需要自动重拨号, 也会重新连接数据中心再发送注册包。下面给了某个厂家的生产的 GPRS DTU 的注册包帧格式。

ID	PHONE_NUMBER	0	IP_ADD	ETX
4 字节	11 字节	1 字节	4 字节	1 字节

数据中心接收到注册包后, 就可以得到 DTU 所设置的 ID 号、所使用的 SIM 卡号、分配的 IP 地址。只有当数据中心接收、分析完注册包后, 才可以确认 DTU 与数据中心建立通信连接, 接下来就可以进行稳定的、可靠地数据传输。

#### 2.3 数据中心通信功能

基于 GPRS 网络的电力监控系统采用客户机/服务器模式, 即 C/S 模式。GPRS DTU 相当一个客户端, 而数据中心就是服务器端。我们可以把 DTU 设置成采用 TCP/IP 协议透明数据传输的工作模式, 数据中心的服务器采用流式套接字。数据中心的通信功能可以通过网络编程来实现。

数据中心服务器端的设计采用被动方式,都是根据 DTU 的状态和上传数据,来作相应的处理。通信部分要完成基本功能如下:

1) 可以建立多个客户端(DTU)与服务器端的通信连接,并可以与各个客户端通信。即一个服务器对应多个 DTU。

2) 管理各个 DTU 与数据中心的建立信道。当某个 DTU 重新拨号,重连数据中心时,要更新信道。

3) 接收从某个 DTU 上传的数据,并区分注册包数据和普通的数据,当接收到注册包数据时,按注册包格式完成分析;接收到普通数据时,按 DTU 与数据中心约定的协议分析数据。

4) 数据中心发送数据到指定的 DTU。

### 3 VC 平台的数据中心实现

数据中心通信功能的设计,是基于 TCP/IP 协议栈的网络编程,采用 C/S 模式。在 VC++6.0 开发环境下,运用 CAsyncSocket 类来实现。这个类在低层次上对 Windows Socket API 进行了封装,可以充分利用 Windows Socket API 编程的灵活性并能满足当有网络事件发生时,利用消息驱动机制处理网络事件。和通常的网络编程的服务器端一样,要构造从 CAsyncSocket 类继承的两个类,一个用于监听客户端的连接请求,在程序中这个类命名为 CServerListen;一个用于与客户端建立通信连接并交换数据,类名为 CDTUClient。数据中心的通信功能的实现除了设计这两个类外,还做其他的处理。

#### 3.1 CDTUClient 类的设计

这个类从 CAsyncSocket 类继承而来的,完成数据中心与 DTU 的通信功能。处理 FD\_READ 事件和 FD\_CLOSE 事件,当接收到 GPRS DTU 发送的数据时,产生 FD\_READ 事件,在其事件处理函数 void CDTUClient::OnReceive(int nErrorCode) 要做数据接收处理。同样地,在 FD\_CLOSE 事件的处理函数 void CDTUClient::OnClose(int nErrorCode) 要做通信链路关闭处理。此外,根据 GPRS DTU 的通信特点,由这个这生成的对象能够区分对应的 DTU 是否已经上传过注册包,完成向数据中心的注册。DTU 当完成向数据中心注册后,这个对象应该根据注册包信息分别把成员数据 m\_strID、m\_strPhone、m\_strIPAddr 和 m\_strLoginTime 设置为 DTU 的设备 ID、SIM 卡号码、动态地址 IP、DTU 登陆注册时间日前,这些设置是在成员函数 SetDTUParam() 中完成的。

这个类的头文件部分代码如下:

```
class CDTUClient : public CAsyncSocket
{
public:
    CDTUClient(CDTUServerDlg *pDlg,int iIndex);
    virtual ~CDTUClient();
    .....
public:
    CString GetLoginTime();
    void SetCheck();
    CString GetPhone();
    CString GetID();
    CString GetIP();
    void SetDTUParam();//设置 DTU 各个参数
private:
    CString m_strLoginTime;
    CDTUServerDlg * m_pDlg;
    bool m_bCheck; //是否已经接收到 GPRS DTU 的注册报文
    int m_iIndex; //在链表中的序号
    CString m_strID;//保存 DTU 的设备 ID
```

```
CString m_strPhone;//保存 DTU 的电话号码
CString m_strIPAddr;//保存 DTU 的动态 IP 地址
```

```
};
```

#### 3.2 多个客户端管理与通信

在 VC 平台下,数据中心的应用程序框架采用对话框模式。具体网络事件的处理函数是对话框类的头文件(.h)中声明,并在对话框类的执行文件(.cpp)中定义。在 CDTUClient 类中的网络事件处理函数中,增加调用对话框类中对应处理函数,这样方便了网络数据把保存到数据库或者显示在界面上,也使得程序结构比较合理,有层次。

在数据中,创建两个链表分别用于管理没有完成注册的和已经完成注册的 CDTUClient 对象。链表定义如下:

```
CPtrList m_ConnDTUList;//用于管理已经完成注册的
CDTUClient 对象
```

```
CPtrList m_NewDTUList;//用于管理还没有完成注册的
CDTUClient 对象
```

CDTUClient 类的数据成员 m\_iIndex 就是用来表明在链表中位置。当有网络事件产生时,可以根据 m\_iIndex 的值在搜寻到链表中产生网络事件的 CDTUClient 对象。正是通过创建数据元素为 CDTUClient 对象的链表,使得数据中心可以准确无误地与多个客户端通信。

在程序设计中,只要一运行数据中心程序,就创建一个 CServerListen 对象用于监听 GPRS DTU 的连接请求。当监听到有 DTU 连接请求,那么就调用 CServerListen 的 FD\_ACCEPT 事件的处理函数 OnAccept。在事件处理函数中,初始化并创建一个 CDTUClient 对象,这个对象标明是没有完成注册操作,即其对象的 DTU 还没有向数据中心发送注册包。同时把这个对象插入一个链表 m\_NewDTUList 中。

部分代码如下:

```
void CDTUServerDlg::OnAccept()
{
    //DTU 客户端连接套接字的标号从 0 开始
    m_pDTUClient =new CDTUClient (this,m_NewDTUList.GetCount());
    if(m_ServerListen.Accept(*m_pDTUClient))
    {
        m_NewDTUList.AddTail(m_pDTUClient);
        .....
    }
}
```

如果数据中心有接收到数据,那么某个 DTU 对应的 CDTUClient 对象就会触发 FD\_READ 事件,并调用其事件处理函数 OnReceive() 函数。所以要在 OnReceive() 函数中区分开是接收 DTU 的注册包数据,还是普通的数据,并要知道是哪个 DTU 上传的数据。当是上传注册包时,接收完整的 21 个注册包数据时,调用 SetDTUParam() 成员函数设置对象的参数,并设置已经完成注册,然后把这个 CDTUClient 对象从 m\_NewDTUList 删除,根据 DTU 设备 ID 号,判断这个 DTU 对应的 CDTUClient 对象是否曾经添加到链表 m\_ConnDTUList 中,若添加过,那么把其设置和原来一样的序号(m\_iIndex 值一样)并删除原来的那个对象,并把这个对象插入到链表 m\_ConnDTUList 中。如果是最新的 DTU 对应的 CDTUClient 对象,那么直接插入到 m\_ConnDTUList 中。如果是普通数据,那么按照数据中心采用的协议规约分析数据,并做相应的处理。

(下转第 26 页)

## 2 VB 实时采集数据实验

本实验采用德国 SPECTRUM 公司生产的 MI.3120 系列 12 位高速 A/D 采集板进行声发射信号的采集。MI.3120 高速 A/D 采集卡支持 PCI 协议, DMA 传输方式和 FIFO, 并提供了设备驱动和 WD2DASKAPI 函数库, 可用于 VC++、VB、Delphi、BC 等能调用 DLL 的编程语言进行开发。本文采用了 VB6.0 开发了基于 Windows XP 操作系统的的多数据采集系统, MI.3120 采集卡提供的 DLL 是 SpcStdNT.dll, 包括 SpcInItPCIBoards, SpcGetParam, SpcGetData16 函数。开发时, 先建立 VB 项目, 在 Modules 中声明 SpcStdNT.dll 及函数名。然后在需要调用处增加对 MI.3120 的注册, 初始化设置、内存分配、触发方式设置等操作, 然后开始采集和数据传输。其中一些关键的函数如下: Function SpcSetParam(ByVal BrdNr As Integer, ByVal RegNr As Long, ByVal Value As Long) As Integer

通过该函数读软件寄存器或状况信息, 对 DMA 控制器初始化, 即完成数传字节数、内存起始地址、工作模式及工作方式等参数的设置。BrdNr 表示想要访问的板卡的索引, RegNr 表示读出的寄存器地址, Value 表示函数调用后的读取参数。

Function SpcGetData16(ByVal BrdNr As Integer, ByVal Channel As Integer, ByVal Start As Long, ByVal Length As Long, ByRef data As Long) As Integer

通过该函数从板卡一个特殊通道读出数据。BrdNr 同上, Channel 表示存储器通道的相关参数, Start 和 Length 表示读出数据的位置和长度, data 表示读出的数据。

以下是用 VB 编写的读取数据部分的源代码:

```
----- get data from board -
For j = 0 To (nMod - 1)
iError = SpcGetData16(BOARDNUM, j, 0, nChPerMod * MEM-
SIZE, nTmpData(0))
For i = 0 To (MEMSIZE - 1) Step 1
    For k = 0 To (nChPerMod - 1) Step 1
        nChData(nChPerMod * j + k, i) = nTmpData(nCh-
PerMod * i + k)
    Next k
Next i
Next j
' ----- paint the signals directly in the form
(quick and dirty) -----
frmMain.Cls
Dim iXOffset, iYOffset, nChOffset As Integer
Dim dWidth, dHeight As Double
' calc settings to show signal between the lines
iXOffset = lineLeft.X1
iYOffset = lineTop.Y1
```

(上接第 24 页)

### 3.3 下发数据

由于电力监控系统的通信模式采用召调式, 所以通信主动权在数据中心。当监控系统需要采集某个子站的电力参数时, 必须主动下发召调报文到指定的子站。数据中心通信的功能中, 可以按照 CDTUClient 类对象中的数据成员 m\_iIndex 在链表中来搜寻到指定的信道, 也可以通过对象中数据成员 m\_strID 来获取指定信道。

根据数据成员 m\_iIndex 获取信道的代码如下:

```
POSITION pos;
CDTUClient *m_pDTUClient;
.....
pos=m_ConnDTUList.FindIndex(iSelected); //其中 iSelected 值
```

```
dWidth=(lineRight.X1- lineLeft.X1)/MEMSIZE
dHeight=(lineBottom.Y1- lineTop.Y1)/(nChannels*4095)
' output signals
For j = 0 To (nChannels - 1)
    nChOffset = 2048 + (4096 * j)
For i = 1 To (MEMSIZE - 1)
    frmMain.Line (iXOffset + dWidth * (i - 1), iYOffset +
dHeight * (nChOffset - nChData(j, i - 1)))- (iXOffset + dWidth *
i, iYOffset + dHeight * (nChOffset - nChData(j, i)))
Next i
Next j
```

图 1 是在 Windows XP 系统下通过 DMA 方式, 采集到的高速 A/D 板的一组时实数据。

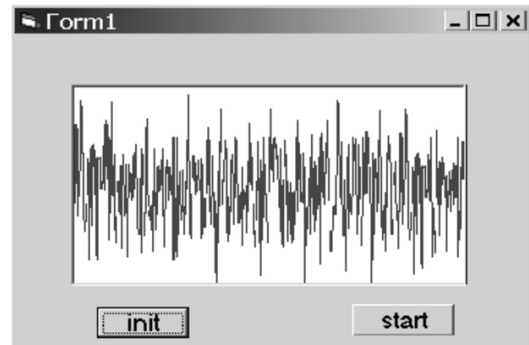


图 1 VB 实时采集

在不同配置的计算机上运行整个数据采集过程用时会有所差别, 但总时间通常为几个毫秒, 磨削直径 1m 的轧辊时, 轧辊旋转一周的时间是几十秒, 在其旋转一周的时间内分布若干个控制周期, 数据采集的速度能够保证在若干个控制周期内控制指令的顺利执行。

### 3 结束语

DMA 数据传输方式由于其速度快、效率高, 是目前高速数据传输方式的主选方法; 本文提到的利用 DLL 传递数组方法解决了我们在数据采集时进行特征参数的提取等问题, 对于在 Win2000/XP 系统下的硬件访问的解决办法也为我们进行现场控制提供了可行的方法。

### 参考文献

- [1] 周明德. 微型计算机系统原理及应用[M]. 北京: 清华大学出版社, 1998
- [2] Evangelous Petroustos, Kevin Hough Visual Basic 5 Developer's Handbook[M]. 北京: 电子工业出版社, 1998

[收稿日期: 2007.9.1]

和 m\_iIndex 一样

```
m_pDTUClient=(CDTUClient *)m_ConnDTUList.GetAt(pos);
m_pDTUClient->Send(pstrBuf,iDataLen);
```

### 4 结束语

经过实际的实验测试, 所设计的 GPRS DTU 的数据中心的通信功能能够很好地完成与多个 GPRS DTU 的通信。在电力监控系统中, 可以满足主站监控系统(数据中心)对下面各个子站的数据采集、控制和设置功能。

### 参考文献

- [1] 叶树华, 高志华. 网络编程实用教程[M]. 北京: 人民邮电出版社, 2006
- [2] Bruce Eckel. C++ 编程思想[M]. 北京: 机械工业出版社, 2002

[收稿日期: 2007.7.24]