

利用泓格产品进行高山无人站监控系统的设计与开发

马玉春 博士 (onspring@sina.com)

1 应用背景

本系统用来监控杭州、温州、宁波等城市的微波站，主控站在城市机房，受控站则在内地或海边的山顶，环境比较恶劣。共有 6 个主控站，每个主控站带 1 个受控站，采用总线型的主控机和受控机两级结构，通过微波信道(图 1 粗黑线所示)在主控机之间传送数据，但在某一时刻只有一台主控机 Polling 所属受控机，被 Polling 受控机作应答处理，其余主控机处于侦听(Listen)状态。主控机 Polling 完所属受控机后，转移 Polling 权，由下一台主控机接管 Polling 权……如此循环。监控内容主要包括柴油发电机、开关电源、市电、蓄电池等的工作状态、环境温度、湿度、防盗等。系统结构如图 1 所示。



图 1 系统结构

2 硬件设计

油机监控器是与大型柴油发电机配套的专用监控器，采用双机热备；1600 与 2800 也是与原设备配套的专用监控器，其中 1600 主要用来监控市电和环境参数，2800 对两组蓄电池(各 12 节)进行监控；报警开关量(遥信)输入模块采用 7041，控制开关量(遥控)采用 7067 模块，受控机的主控模块则选用 7188。7188 有四个串行口，其中两个 RS232(仅含 Rx、Tx、GND 信号线)，一个 RS485，一个 9 针全信号 RS232 口(通过内部跳线，可将此口改变为 RS485 口)。7188 模块接口多，抗恶劣环境，同时，提供丰富的内部函数，有利于快速开发出性能可靠的下位机监控系统程序。受控站系统的硬件结构见图 2 所示。由于各监控模块均使用 RS485 接口，所以需要两个 RS232/RS485 转换接口。COM1 口为全信号 RS232，连接专用 Modem，从而进入微波信道进行数据的传输。主控站通过 RS232 接口与专用 Modem 连接，进入微波信道。

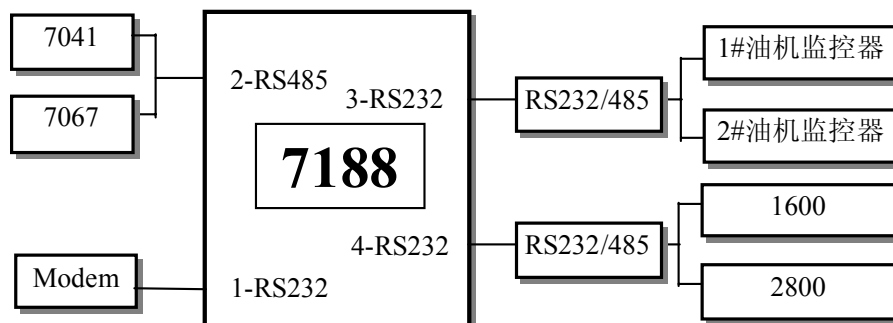


图 2 受控站硬件结构

3 软件实现

3.1 主控机部分

主控机部分采用组态王 5.1 来实现。该系统可运行于 Win98/NT 平台，真正 32 位程序，

实时多任务、多线程，采样速度快，可靠性高，同时，支持分布式历史数据库及拨号网络，具有丰富的 ActiveX 控件，广泛应用于钢铁、化工、环保、国防、航空航天等重大领域。

利用组态王开发实时计算机监控系统方便快捷。首先，定义设备，加载通讯驱动程序。如对于本系统，是利用 RS232 接口来与受控机通讯的，其定义设备时选取的顺序如下：智能模块—I-7000 系列—I-7188—Modbus RTU Extention。然后，再规定 RS232 的工作参数。其次，定义数据词典，规定变量名、数据类型、寄存器之间的关系。数据词典中的变量可以根据原始数值自动计算所对应的线性目标值。最后，根据组态王提供的图元控件设计用户界面，无论是字符串还是图元，均可建立“动画链接”，从而，将数据变量和数据的显示结合起来。数据显示包括两种形式，一是数据的文本格式的显示，二是数据的动画表示，如对于报警量，正常显示绿灯，报警显示红灯等。如此三步，主控机软件开发基本完工。

3.2 受控机部分

3.2.1 受控机主程序设计

7188 提供了丰富而方便的库函数，其中包括关键的看门狗、相关的数据发送和接收函数、时钟中断函数等等，同时，还提供了丰富的例程，因而，使得受控机软件逻辑简捷，开发极其迅速可靠，与其它厂家的模块也能极其友好地连接。7188 的一个串口控制两台智能设备，这是通过地址来区分的。对于泓格产品的 7000 系列模块，其工作参数及模块地址是可调的，可在系统组装之前对其进行设置。而其它智能模块的工作参数和地址则是由厂家设定的，不可更改。

由于 7188 内嵌基本的 DOS 环境，因而，采用 TC2.0 来编写受控机程序。主程序首先初始化系统，包括对各串口的初始化及系统运行参数的初始化，并安装用户时钟中断。由于该用户中断固定为 1ms，因而，在中断程序中应避免接收和发送数据，因为监控系统中的波特率一般不超过 9600(太高可能导致数据传输的不可靠)，而对于 9600 的波特率，传输 1 个字节的数据大约需要 1ms，一般情况下，所选择的波特率低于 9600，而且，一次发送和接收的数据一般超过 2 个字节，因而，会引起 DOS 重入，从而使系统不可靠。所以，时钟中断程序应尽量简捷，主要完成各个工作节拍之间的协调问题，主要工作在主程序中进行，即轮询时间或节拍到，即向智能模块或设备发送查询命令，而当有数据返回时，再去读取数据并处理。其软件系统框图如图 3 所示。

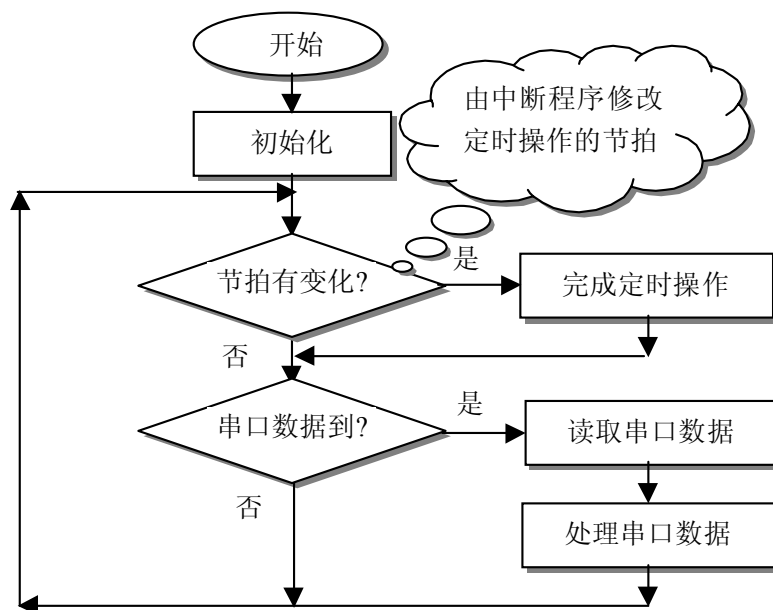


图 3 受控机主程序框图

3.2.2 通讯问题及解决方案

对于计算机监控系统，最关键的问题是数据通讯的可靠性。泓格产品的 I/O 设备数据传输平稳，能与 7188 可靠地连接，但其它厂家模块的数据传输并不象所想象的那么连续，因而，应该根据所接收的数据的不连续性作相应的等待，时间太短，数据将可能丢失，时间太长，则可能造成死机或将下一批的数据也一块接收进来。数据接收程序见算法 1 中的 ReadDataDelay 子程序。经过反复调试，对于本系统，发现延时数据大于 0x0140 即可，小于 0x0130 则数据错误率达 1/4 左右，并且，主要是每次丢失 1~2 个字节的数据。0x0130 与 0x0140 之间为临界值。数据发送也用一个统一的子程序来完成，见算法 1 中的 SendData 子程序，数组中第一个字节用来存放收到的字节数，其后为数据内容。以下算法对嵌入式模块的程序设计具有通用性，其基本原理可以广泛使用。

```

/*****
/*      ReadDataDelay:读取串口 nPort 中的数据，放入      */
/*          bData 中，nDealy 为需要等待的节拍数          */
*****/
void ReadDataDelay(int nPort, unsigned char *bData, int nDelay)
{
    int nData=0; /*接收到的字节数*/
    int iCount=0; /*等待计时*/
    while(iCount < nDelay)
    {
        if(IsCom(nPort)==QueueIsNotEmpty) /*nPort 口有数据*/
        {
            nData++;
            bData[nData] = ReadCom(nPort); /*读取一个字节的*/
            iCount = 0; /*有数据，计时清零*/
        }
        else iCount++; /*没有数据，则等待计时*/
    }
    /*bData 数据中第一个字节地址存放收到的字节数，其余为字节内容*/
    bData[0] = nData;
    ClearCom(nPort); /*清除串口数据内容，为下次数据接收作准备*/
}

/*****
/*      SendData: nPort 为串口号，ComData 为需要发送的数据      */
*****/
void SendData(int nPort, unsigned char *ComData)
{
    int nlength, i;
    nlength = ComData[0];
    if(2 == nPort) Set485DirToTransmit(nPort);
    for(i=0; i<nlength; i++)
    {
        ToCom(nPort, ComData[i+1]);
    }
}

```

```
        if(2 == nPort) WaitTransmitOver(nPort);
    }
    /*Com2 为 485 口，应调整其数据传输方向*/
    if(2 == nPort) Set485DirToReceive(nPort);
    return;
}
```

算法 1 数据的接收和发送

4 调试及结论

泓格产品的 7000 系列组合方便，而且，7188 有四个 RS232/485 串口，价格却非常便宜，一块 7188 主控模块只要 2000 多元，且同样抗恶劣环境，曾用电吹风对其连续劲吹，使其表面发烫（温度在 50℃ 以上），却依然可靠地工作。由于上位机提供了组态软件，因而，上位机程序开发迅速，性能可靠；下位机提供了丰富的库函数和例程，使得下位机软件的逻辑简捷，编程灵活而迅速，看门狗功能又能使其自动恢复。

作者长期从事计算机监控方面的项目与研究，设计了“通用多功能计算机监控系统测试软件”，专门用来通过 RS232 或 RS485 测试监控系统或相应的智能设备^[1]。该软件既能充当受控机用来测试主控机程序，又能充当主控机用来测试受控机程序，并在多个监控系统中得到验证。在测试过程中发现，7000 系列的模块数据传输平稳，并且，连续测试 48 小时（5 秒测试 1 次），非泓格产品的监控器平均发生 25 次数据传输错误，而对于 7000 系列的模块却没有一次错误记录；用该测试软件模拟受控机对主控机（组态软件）进行测试，也没有错误记录。目前，该系统已经在实验室调试通过，效果良好。由于采用了泓格产品的组态软件及 7000 系列的模块，该系统开发迅速，性能可靠，因而，极有推广价值 (onspring@sina.com)。

参考文献：

[1] 马玉春, 宋瀚涛. 串行通讯协议的研究与应用. 计算机应用研究, 2004.4, p228