

数据库在测控系统中的应用

摘要: 本文探讨了在测控系统中引入上位机, 建立数据库的新思路, 分析并使之具体实现。其优势在于能很好的保存原始测控资料, 并可对数据进行分析统计等, 对提高测控质量有重要意义。此外, 可充分利用数据库的优势, 延伸功能, 如对技术水平、工作量进行考核等。

关键词: 测控系统 数据库 Visual Foxpro Visual C++ 协议

张大兴 (杭州电子工业学院计算机学院 310037)
翁文庆 (浙江财经学院计算机学院 310037)



测控系统是机电一体的复杂系统, 多数的“电”部分又是由单片机或专用的功能电路来实现。众所周知, 它们的局限性在于存储容量小、编程语言功能较差, 编程效率低, 数据难以保存。在传统测控系统中引入微机作为上位机, 充分利用数据库管理系统的优势可克服以上不足。

1 数据库支持测控系统需解决的主要问题

1.1 和一般 MIS 系统的比较

通常的 MIS 系统, 由数据录入、查询、统计、打印等。而测控系统中的数据库其基本数据主要来自下位机的实时采集。系统应具有从下位机接收数据的能力, 并能实时的把接收或发送的数据显示出来。

数据的发送或接收涉及计算机通信, 这通过对 I/O 口的读写来实现。数据库系统没有直接对端口访问的能力。

在多路测控系统的情形下, 数据库如何并行控制各系统也是一个需周密考虑的方面。

1.2 和一般测控系统的比较

新的测控系统应能接受上位机的指令, 并进行相应的操作。这要求测控系统和上位机之间有一个很好的通信协议, 保证指令的正确执行, 保证数据的正确传输。

此外, 虽然引入了数据库系统, 仍应保证系统有独立的测控功能。

2 编程开发

Visual C++ 有很好的对底层系统的编程能力, 本文中用它设计开发对端口的操作函数, 编译成动态链接库供调用, 以下简单介绍端口操作函数。

2.1 端口操作函数

(1) 初始化端口

```
extern "C" void PASCAL EXPORT portinit(int *i_result,
int comnumber )
```

参数 i_result 为初始化的返回值
comnumber 端口号

初始化端口调用以下函数实现:

```
int _outp ( unsigned short port, int databyte )
```

以初始化 COM1 口为例, 可通过向端口写以下数据实现初始化

```
_outp(0x3fb,0x80);
_outp(0x3f8,0x18);
_outp(0x3f9,0x00);
_outp(0x3fb,0x03);
_outp(0x3fc,0x0b);
_outp(0x3f9,0x00);
```

更详细的信息可看文献。

(2) 读端口数据

```
extern "C" unsigned int PASCAL EXPORT infromport(
int *i_in , int comnumber )
```

参数 i_in 为从端口读入的数据
comnumber 端口号

读端口调用以下函数实现:

```
int _inp( unsigned short port );
```

(3) 写端口数据

```
extern "C" void PASCAL EXPORT outtoport(int *i_out
,int comnumber)
```

参数 i_out 为向端口输出的数据

comnumber 端口号

(4) 端口延时

```
extern "C" void PASCAL EXPORT timedelay( int i_in ,
int comnumber )
```

参数 i_in 为端口操作前的延时秒种数

comnumber 端口号

在具体开发时要注意端口可能出现的各种情况,对数据传输过程中可能的错误也应有相应的防范。

2.2 数据类型的转换

在数据传输过程中,应根据数据的类型采用不同的编码方案,减少传输量、减少传输中的出错,提高效率。数据量采用压缩BCD码,其他用二进制编码来传输。此外,用三个ASCII码来表示一个浮点数,也存在转换的问题。

Visual Foxpro 是目前流行的中小型数据库,对系统的要求不高,有利于降低整体成本。它的特点是有自含的用户界面编程语句,可以满足测控系统界面设计的要求。Foxpro 有一定的字位处理能力,在数据转换上比较容易实现,它也提供了调用外部函数的接口,可以借此控制端口,读写数据。涉及数据转换的函数主要有:

(1) BCD 码转换函数

(2) 浮点数到 ASCII 码的转换

```
float_c(x,@a,@b,@c)
```

参数: X 是被转换的数;

a,b,c 是转换后的三个字符。

(3) ASCII 码到浮点数的转换

```
c_float(a,b,c,@x)
```

参数: a,b,c 是用于表示浮点数的三个字符

X 是转换后的浮点型实数

需要特别提醒的是测控系统经常涉及有效数字的概念,而数据库是没有这个概念的,在数据转换和保存时应注意相应的精度控制。

2.3 外部函数的调用

在数据库系统主程序中说明外部函数

```
DECLARE INTEGER PORTINIT IN PORTDLL
```

```
INTEGER @ , INTEGER
```

```
DECLARE INTEGER OUTTOPORT IN PORTDLL
```

```
INTEGER @ , INTEGER
```

```
DECLARE INTEGER INFROMPORT IN PORTDLL
```

```
INTEGER @ , INTEGER
```

```
DECLARE INTEGER TIMEDELAY IN PORTDLL
```

```
INTEGER , INTEGER
```

这些就是用 Visual C++ 编写的用于端口操作的动态连接库函数。

3 测控系统实例

笔者参与开发的膜式煤气表校验测控系统,就引入了微机数据库,图1是简单的示意图。它是一个多路测控系统,最多可以同时有二十个测试台工作,每个测试台又可同时测试若干煤气表。每个测试台可在不同的工作状态,例如不同的流量点。

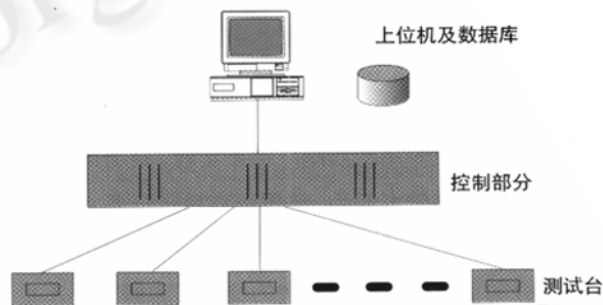


图 1 膜式煤气表测控系统示意图

本系统中上位机的工作流程为:

第一步: 工作环境设置。系统最多可同时允许对二十个测试台进行测试控制,但实际工作中可能仅有少数几个处于工作状态,为加快轮询和数据处理速度,可在开始工作前设置相关测试台的工作状态。

第二步: 通信端口初始化。检测系统端口是否处于完好状态,设置通信速率,在成功后返回相应的信息。

第三步: 轮询各个测试台。上位机询问测试台有无新的测试数据可供读取,具体取决于相关的系统及协议。在膜式煤气表测试系统中包括几个子步骤。

(1) 上位机连续发送 4 个字节的 "0"(本系统所有的测试项目中,不会出现连续 4 个字节为 "0" 的情形)作为特殊的控制字符,通知测试台,上位机准备读取数据,测试台做好响应指令的准备。

(2) 上位机发送测试台编号,测试台接收到后,比较是否和本身的编号相符合。对应的测试台做出响应:有或无测试数据可供上位机读取(若测试台中途停止工作,也即不在工作状态,则没有任何响应,此时上位机的超时处理机制发挥作用)其他测试台则不作任何响应。

(3) 若有数据,则读取测试数据;测试台无响应超时或回答无数据,则对下一个测试台进行轮询。

第四步: 测试数据的处理。由于和下位机的通信是以字节为单位进行的,采用多种编码形式以减少通信的数据量,上位机需要对接收到的数据进行一定的分析处理,并保存到数据库中。上位机和下位机的通信协议规定: 每一个测试项均采用以三个字节表达的浮点数形式来传输,每一个测试台的通信数据的长度和格式是固定不变的,接收方收到数据后必须向发送方回送同一数据(特殊控制信号除外),以确认数据在传输过程中没有出错! 数据传输出错时,回到第三步重新开始。数据成功传输后,测试台删除已传输的数据。

第五步: 若继续测试,则重新进行第三步操作; 否则,结束测试。

实践表明,本系统可允许不同的测试方式同时工作,数据库能及时完成数据传输、转换、显示、保存等工作; 系统有很好的灵活性,测试台可中途加入或退出测试工作; 系统有很好的传输可靠性,加上有一个严密的通信协议,数据的出错概率几乎为零; 对原始测试数据的保存分析,可以很快得出导致误差的主要原因,提高了产品质量。此

外,利用数据库在数据管理上的便利,对测试员的工作量,工作效果可进行考核,对产品检验责任也有了很明确的归属。该系统很好的满足了实际使用要求。

4 结束语

传统的测控系统中大量的测试及控制数据限于系统的资源白白丢弃。本文探讨了在测控系统中引入上位机,建立数据库的新思路,分析并使之具体实现。其优势在于能很好的保存原始测控资料,并可对数据进行分析统计等,对提高测控质量有重要意义; 可充分利用数据库的优势,延伸功能,如对技术水平、工作量进行考核等; 可简化控制电路设计,部分控制功能可由微机承担; 投入小,性能价格比明显,有利于测控系统上档次,增加竞争力。

参考文献

- 1 陈宗兴等 编著,《Visual Foxpro 6.0 中文版编程经验和技巧》,青岛出版社,1999。
- 2 朱岩、高伟等编著,《标准Windows程序设计》,人民邮电出版社,1997。
- 3 朱慧真编著 《80x86 汇编语言教程》,清华大学出版社,1995。